New Methods to Improve the Pixel Domain Steganography, Steganalysis, and Simplify the Assessment of Steganalysis Tools

By
Omed Saleem Khalind

The thesis is submitted in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy of the University of Portsmouth

Supervised By
Dr. Benjamin Aziz

School of Computing
University of Portsmouth
Lion Terrace, Portsmouth, Hampshire
PO1 3HE, United Kingdom

December 2015

Copyright

ABSTRACT

Unlike other security methods, steganography hides the very existence of secret messages rather than their content only. Both steganography and steganalysis are strongly related to each other, the new steganographic methods should be evaluated with current steganalysis methods and vice-versa. Since steganography is considered broken when the stego object is recognised, undetectability would be the most important property of any steganographic system. Digital image files are excellent media for steganography, as they have redundancy in their representation. Also, the most widely used method of image steganography is the least significant bit (LSB) embedding.

This thesis investigates the latest methods of pixel domain steganography and provides new efficient approaches to improve them in three perspectives: embedding, detection, and the digital forensics investigation process. Firstly, the probability of detection is considered for non-adaptive LSB and 2LSB image steganography even for the embedding rate of 1. The proposed method noticeably reduced the probability of detection for different detection methods via improving the embedding efficiency of both LSB and 2LSB methods, which is not restricted to a specific steganalysis attack.

The extensions to LSB steganography methods have received great attention from steganographers, especially 2LSB, because it is easy to implement, has a higher capacity, is visually imperceptible, brings complex changes to the image pixel values and is harder to detect. The proposed method improves the detection accuracy of the current state of the art targeted 2LSB steganalysis methods via a novel approach pixel value grouping and statistical analysis of the image pixel values histogram. Moreover, a discrete classifier version of the proposed method is developed which gives a label (‘Stego’ or ‘Clean’) to the analysed image and avoids the overhead of setting a right threshold value.

The last perspective of this research considers the evaluation process of the steganalysis tools and simplifying the digital forensics investigation process. Hence, a novel statistical method is proposed to effectively simplify the investigation process by showing the area of differences between the testing image set and the random set of images that is used as a baseline. It also indicates whether the difference is significant or not.

All the above mentioned novel approaches included in this thesis are proven, in both theoretical and practical perspectives, to be better than the current state-of-the-art methods and add some value to the knowledge in the field of steganography, steganalysis and its applications.

Key words: Steganography, Steganalysis, LSB embedding, 2LSB embedding, Forensic steganalysis, LSB embedding, 2LSB steganalysis

CONTENTS LIST

DECLARATION

Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award.

Signature

Omed Saleem Khalind

December 2015

Total word count: 49532

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ALE	Amplitude of Local Extrema
ASIRRA	Animal Species Image Recognition for Restricting Access
AUC	Area Under the Curve
BPCS	Bit-Plane Complexity Segmentation
CF	Characteristic Function
DCT	Discrete Cosine Transform
DIH	Difference Image Histogram
DWT	Discrete Wavelet Transform
ENMPP	Expected Number of Modifications Per Pixel
EOF	End Of File
EPoV	Extended Pairs of Values
EXIF	Extended File Information
FN	False Negative
FP	False Positive
GUI	Graphical User Interface
HAS	Human Auditory System
HCF	Histogram Characteristic Function
HVS	Human Visual System
JPEG, JPG	Joint Photographic Experts Group
LSB	Least Significant Bit
LSM	Least Squares Method
MLSB	Multiple Least Significant Bits
MSE	Mean Square Error
MSSIM	Multi-Scale Structural Similarity
PDF	Probability of Density Function
PoV	Pair of Value
PRNG	Pseudo-Random Number Generator
PSNR	Peak Signal to Noise Ratio
RGB	Red, Green, and Blue
ROC	Receiver Operating Characteristic
RS	Regular and Singular
SDCS	Sum and Difference Covering Set

SM2LSB	Single Mismatch two Least Significant Bits
SMLSB	Single Mismatch Least Significant Bit
SP	Sample Pair
SPA	Sample Pair Analysis
SSIM	Structural Similarity
TN	True Negative
TP	True Positive
VIF	Visual Information Fidelity
VSNR	Visual Signal to Noise Ratio
WS	Weighted Stego

ACKNOWLEDGEMENT

Firstly, I would like to express my sincere gratitude to my first supervisor Dr. Benjamin Aziz for his continuous support of my PhD study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all stages of research and writing this thesis. I could not have imagined having a better advisor and mentor for my PhD study.

Besides my supervisor, I would like to thank my advisors Dr. Carl Adams and Dr. Julio Hernandez-Castro, for their insightful comments and encouragement, but also for their hard questions that encouraged me to widen my research from various perspectives.

My sincere thanks also to all other research staff members and the Head of the School of Computing, University of Portsmouth, who provided me with an opportunity to gain many research skills via attending their seminars, joining many research activities and supporting me to attend some conferences. I would also like to thank the Graduation School of the University of Portsmouth for their valuable Development Programs. Without their precious support it would not be possible to conduct this research.

I would like to thank my fellow PhD students as well in the school for the stimulating discussions, sharing information, and for all the fun we have had in the last four years.

Last but not the least, I would like to thank my family: my wife, my parents, my sisters and brothers for supporting me spiritually throughout writing this thesis and my life in general.

DISSEMINATION

The following papers have been published as a direct result of the research discussed in this thesis.

Journals

- **Khalind, O. S., Hernandez-Castro, J. C., & Aziz, B. (2013). A study on the false positive rate of Stegdetect. *Digital Investigation, 9*(3–4), 235-245.**

In this paper we analyse Stegdetect, one of the well-known image steganalysis tools, to study its false positive rate. In doing so, we process more than 40,000 images randomly downloaded from the Internet using Google images, together with 25,000 images from the ASIRRA (Animal Species Image Recognition for Restricting Access) public corpus. The aim of this study is to help digital forensic analysts, aiming to study a large number of image files during an investigation, to better understand the capabilities and the limitations of steganalysis tools like Stegdetect. The results obtained show that the rate of false positives generated by Stegdetect depends highly on the chosen sensitivity value, and it is generally quite high. This should support the forensic expert to have better interpretation in their results, and taking the false positive rates into consideration. Additionally, we have provided a detailed statistical analysis for the obtained results to study the difference in detection between selected groups, close groups and different groups of images. This method can be applied to any steganalysis tool, which gives the analyst a better understanding of the detection results, especially when he has no prior information about the false positive rate of the tool.

Conferences

- **Khalind, O., & Aziz, B. (2013, 12-15 Dec. 2013). *Single-mismatch 2LSB embedding steganography.* Paper presented at the International Symposium on Signal Processing and Information Technology(ISSPIT), 000283 – 000286.**

This paper proposes a new method of 2LSB embedding steganography in still images. The proposed method considers a single mismatch in each 2LSB embedding between the 2LSB of the pixel value and 2-bits of the secret message, while the 2LSB replacement overwrites the 2LSB of the image's pixel value with 2-bits of the secret message. The number of bit-changes needed for the proposed method is 0.375 bits from the pixel values of the cover image, which is less than the

0.5 bits of 2LSB replacement. It also reduces the effect of 2LSB embedding pattern of change, which results in lower probability of detection by 44% according to the experimental results.

- **Khalind, O., & Aziz, B. (2014).** ***Detecting 2LSB steganography using extended pairs of values analysis.*** **Paper presented at the Mobile Multimedia/Image Processing, Security, and Applications 2014, 9120, 912003-12.**

In this paper, we propose an extended pairs of values analysis to detect and estimate the amount of secret messages embedded with 2LSB replacement in digital images based on chi-square attack and regularity rate in pixel values. The detection process is separated from the estimation of the hidden message length, as it is the main requirement of any steganalysis method. Hence, the detection process acts as a discrete classifier, which classifies a given set of images into stego and clean classes. The method can accurately detect 2LSB replacement even when the message length is about 10% of the total capacity, it also reaches its best performance with an accuracy of higher than 0.96 and a true positive rate of more than 0.997 when the amount of data are 20% to 100% of the total capacity. However, the method puts no assumptions neither on the image nor the secret message, as it tested with two sets of 3000 images, compressed and uncompressed, embedded with a random message for each case. This method of detection could also be used as an automated tool to analyse a bulk of images for hidden contents, which could be used by digital forensics analysts in their investigation process.

- **Khalind, O., & Aziz, B. (2015).** ***LSB Steganography with Improved Embedding Efficiency and Undetectability.*** **Paper presented at the The Fourth International Conference on Signal & Image Processing (SIP 2015), 89 – 105, Zurich, Switzerland.**

In this paper, we propose a new method of non-adaptive LSB steganography in still images to improve the embedding efficiency from 2 to 8/3 random bits per one embedding change even for the embedding rate of 1 bit per pixel. The method takes 2-bits of the secret message at a time and compares them to the LSBs of the two chosen pixel values for embedding, it always assumes a single mismatch between the two and uses the second LSB of the first pixel value to hold the index of the mismatch. It is shown that the proposed method outperforms the security of LSB replacement, LSB matching, and LSB matching revisited by reducing the probability of detection with their current targeted steganalysis methods. Other advantages of the proposed method are reducing the overall bit-level changes to the cover image for the same amount of embedded data and avoiding complex calculations. Finally, the new method results in little additional distortion in the stego image, which could be tolerated.

- **Khalind, O., & Aziz, B. (2015).** ***A better detection of 2LSB steganography via standard deviation of the extended pairs of values.*** **Paper presented at the Mobile Multimedia/Image Processing, Security, and Applications 2015, 94970E-8, Baltimore, Maryland.**

This paper proposes a modification to the Extended Pairs of Values (EPoV) method of 2LSB steganalysis in digital still images. In EPoV, the detection and the estimation of the hidden message length were performed in two separate processes as it considered the automated detection. However, the new proposed method uses the standard deviation of the EPoV to measure the amount of distortion in the stego image made by the embedding process using 2LSB replacement, which is directly proportional with the embedding rate. It is shown that it can accurately estimate the length of the hidden message and outperform the other methods of the targeted 2LSB steganalysis in the literature. The proposed method is also more consistent with the steganalysis methods in the literature by giving the amount of difference to the expected clean image. According to the experimental results, based on analysing 3000 never-compressed images, the proposed method is more accurate than the current targeted 2LSB steganalysis methods for low embedding rates.

CHAPTER 1: INTRODUCTION

1.1 Overview

Steganography is the art and science of hiding communication by embedding secret data in public cover media without raising suspicion. Due to the availability of the Internet, lack of trust, and the demand for secret communication, people try to secure their private messages using more advanced steganography techniques rather than traditional cryptographic methods. The idea of hiding secret messages in multimedia files like images and video gives an opportunity to a variety of application areas beyond steganography, collectively known as information hiding. Any digital media with some redundancy in their representation could be used by steganographers for embedding secret messages. Hence, digital images became one of the most common cover media used for this purpose. Also, the most widely used method of steganography is least significant bit (LSB) replacement in digital images, due to its extremely easy implementation, imperceptibility, and reasonable capacity. However, LSB steganography is very easy to attack and there are many methods in the literature that can accurately detect them.

Many studies consider imperceptibility to be the most important property of steganography (Al-Mohammad, 2010), but this research considers undetectability, because nowadays almost all steganographic methods generate imperceptible stego objects, but they are still detectable by the statistical methods of steganalysis. Moreover, any steganographic algorithm is considered broken when the stego media is recognised, even if the secret message itself is not recovered (Böhme & Westfeld, 2004). Thus, undetectability is essential and is considered as the most important property of steganography.

The hard truth for steganographers is that LSB method is reliably detectable by current steganalysis methods with a very accurate estimation of the length and the embedding locations of the secret message. Thus, the modified versions of LSB embedding became the field of interest by steganographers in the last few decades (Luo, Liu, Yang, Lian, & Zeng, 2012; Yang, Liu, Luo, & Liu, 2008; Xiaoyi Yu & Babaguchi, 2008; Xiaopi Yu, Tan, & Wang, 2005). Also, there was a growth of interest of using the extensions of LSB steganography, such as using more than one LSB for data embedding. More specifically, the 2LSB steganography as it is could be implemented very easily, is visually imperceptible, has higher capacity than LSB steganography, and also results in more complicated changes in the pixel values, which makes it harder to detect. However, they are again detectable by the current steganalysis methods.

One of the most realistic applications of steganalysis methods is its usability as a detection tool for hidden contents by digital forensics analysts, especially for cases that related to cybercrime, child pornography, and terrorist activities. In this case, as the embedding methods are continuously improved, the digital forensics analyst also needs better detection methods to reduce the probability of false alerts in their investigation process.

Thus, this thesis considers steganography in three perspectives: embedding, detection and its applications. Firstly, from the steganographic point of view, this research focuses on improving the embedding efficiency and reducing the probability of detection for both LSB and 2LSB image embedding methods even for the embedding rate of 1. Secondly, from the steganalysis point of view, this research concentrates on improving the detection accuracy of 2LSB image steganography, especially for low embedding rates. The last point of view is the application of steganalysis methods; this research proposes a statistical method to be applied on the detection results. This method can be used for evaluating the steganalysis tools and also helping the digital forensics analysts in their investigation process while analysing a bulk of images to show the area of differences between two samples of images. This approach helps the forensics analyst by narrowing down the investigation process to the area of interest and neglecting the insignificant parts of the detection results.

1.2 Motivations and Research Problem

There has been an explosive growth in image steganography, steganalysis, and their applications in the past few years, particularly in signal processing (Fridrich, 2009). Images are excellent media for steganography due to having redundancy in their representation, and the most widely used method of image steganography is LSB embedding.

There are two main types of LSB steganography, adaptive and non-adaptive methods. The adaptive method takes the content of the image into consideration and leaves some parts of the image unmodified. This method results in less probability of detection, but with less capacity, and it differs from one image to another. The non-adaptive method embeds data into the image regardless of its content. This embedding method usually has a higher capacity with higher probability of detection.

However, there are few opportunities to improve embedding efficiency for non-adaptive methods of LSB, especially when all pixel values are involved in the embedding process (in other words, when the embedding rate is 1). Also, most detection methods of the extended methods of LSB

embedding belong to universal steganalysis methods (Luo et al., 2012; Yang et al., 2008; Xiaoyi Yu & Babaguchi, 2008; X. Yu et al., 2005), and there are few targeted methods, especially 2LSB image steganography, which are more common but not very accurate for low embedding rates.

Moreover, from the application point of view, most steganalysis methods give the probability of stego class membership instead of giving labels ('Clean' or 'Stego') to the analysed image. Hence, choosing the right threshold value is left to the analyst, which could be very challenging by having a direct effect on the rate of false alarms (false positives and false negatives). This could be even more essential and problematic for digital forensic analysts involved in analysing a bulk of images. Thus the evaluation of steganalysis methods is very complex and the digital forensics analyst has a very wide area for investigation. The literature lacks having an easy and efficient method to narrow down the investigation process in relation to steganalysis tools.

In this research, the embedding efficiency of both LSB and 2LSB embedding of non-adaptive image steganography are improved, even for the embedding rate of 1, by introducing the concept of single mismatch embedding method. In addition to improving the embedding efficiency it reduces the overall bit-level cost of pixel value changes and results in lower probability of detection by the current steganalysis methods.

Regarding the detection of the extended methods of LSB steganography, 2LSB steganography proposes a new method in two different forms: as a discrete classifier that does not need the threshold value to be set, and directly giving the label to the analysed image ('Clean' or 'Stego'). Also as a probabilistic classifier that gives the probability of the stego class membership, or the length of the embedded secret message. This method outperformed the current targeted steganalysis methods of 2LSB embedding, especially for low embedding rates.

From the digital forensics point of view, as a realistic application, an efficient statistical method is proposed to process the detection results for identifying the significant parts of it for further investigation and neglecting the insignificant ones. This statistical approach could also be used to evaluate a certain steganalysis method.

1.3 Research Aim

The motivation and research problem section stated that it is a three-fold research: steganography, steganalysis and its application. This thesis proposes some innovative non-adaptive methods, called single mismatch, to improve the embedding efficiency with less bit-level cost of pixel value changes for both LSB and 2LSB image steganography, and reduce the

probability of detection by the current targeted steganalysis methods. This method can be applied even if the embedding rate is 1.

It also proposes a new method of detecting 2LSB embedding of digital image steganography, called extended pairs of values, in such a way that outperforms the current targeted 2LSB steganalysis methods in terms of detection accuracy and usability as a discrete binary classifier. It can also maintain its accuracy for low embedding rates. This is because, instead of the probabilistic model of the clean image, it relies on the arithmetic mean of the frequency of occurrences in each extended pair of values, which stay unmodified before and after the embedding process has taken place.

The third aim of this research is proposing an easy and efficient statistical method to simplify the evaluation process of the steganalysis methods by people like digital forensics analysts using steganalysis methods as a tool (black box). This could be done by applying more than one steganalysis method and comparing the statistical results on the same set of images. Also, to show the area of differences between the random set of images as a baseline and the testing set of images. This would be very useful as it simplifies the investigation process by specifying the area of interest (significant area of differences) for further investigation in order to reduce time, cost and complexity.

1.4 Contributions

The contributions of this thesis could be classified into three main areas in relation to information hiding and detection. The first contribution is the development of a novel non-adaptive embedding method that can be applied in both LSB and 2LSB steganography. This embedding method, in both cases, can be applied even if the embedding rate is 1, without having any restrictions on the cover image and the saturated pixel values.

As shown in chapter four, the proposed embedding method (single mismatch) improves the embedding efficiency (ENMPP) of LSB replacement from 0.5 to 0.375 pixels per message bit and the 2LSB replacement from 0.75 to 0.687 pixels per two message bits. In addition to the embedding efficiency, the proposed method reduces the bit-level cost of pixel value changes that directly affects the probability of detection by methods relying on the binary similarity measures in pixel values. The single mismatch embedding method also remarkably reduces the probability of detection by current targeted steganalysis methods without restricting it to a certain type of steganalysis attack (e.g. histogram attack).

All the above mentioned advantages of the proposed method (single mismatch) are theoretically and experimentally proven in chapter four. The only limitation with this method is that it results in slightly lower value of Peak Signal to Noise Ratio (PSNR) compared to other methods in the literature. However, this amount (nearly 1.7 dB) can be neglected as it still stays very close to other methods and very far from the lower limit of the PSNR value discussed in chapter four.

The second contribution is the detection method of 2LSB image steganography, which is applied as discrete and probabilistic classifier. This method relies on a new grouping scheme of image pixel values by considering the characteristics of the 2LSB embedding method in relation to pixel value transitions. This method is named as extended pairs of values, or EPoV for short. EPoV outperforms the current targeted 2LSB steganalysis methods in the literature in terms of detection accuracy and the estimation of the hidden message length.

This method, as shown in chapter five, firstly takes the application of digital forensics analysis into consideration. Hence, it mainly focuses on giving labels to the analysed images (‘Clean’ or ‘Stego’) rather than the membership probability to the stego class. In this case, the overhead of setting a right threshold value is eliminated and the method is thus a ready to use detection tool. Therefore, two sets of compressed and uncompressed images are used in the experiment, not only considering the ‘in laboratory’ conditions. This method can maintain its accuracy even for small embedding rates (0.6 for the embedding rate of 0.05, 0.879 for 0.1, and 0.962 for 0.2), where the embedding rate is the percentage of the cover image pixel values involved in the embedding process. These classification results are also shown as a Receiver Operating Characteristic (ROC) graph in the results section in the same chapter.

Probabilistic classification is considered (again discussed in chapter five) in calculating the probability of stego class membership from the analysed image. This detection method combines the new grouping scheme with the standard deviation of the intensity histogram. The value of the standard deviation of the histogram of EPoV, after subtracting 1 or the expected value for clean images, ranges from 0 to 0.5, which implies the amount of the image portion changed by 2LSB embedding method.

This method outperforms the current targeted 2LSB steganalysis methods in terms of detection accuracy even for very low embedding rates, because it relies on the arithmetic mean of the histogram of the EPoV, which stays the same before and after the 2LSB embedding process.

The third contribution is the novel statistical approach for analysing the results of steganalysis tools (methods). This can be used for two different purposes, the evaluation of steganalysis

methods and simplifying the investigation process of digital forensics analysis. The evaluation process can be achieved by applying two different steganalysis methods on the same set of clean and stego images and statistically analysing their results to see whether the difference between the results are significant or not.

Simplifying the digital forensics investigation process is performed by applying statistical method on the detection results of two sets of images, the first of which consists of a bulk of random images to be used as a baseline for comparison, and the second of which is the testing set selected for investigation by digital forensics analysis. The proposed method shows the area of differences and specifies whether the difference is significant or not. Significant difference is determined if the digital forensics analyst can focus on this area and do further investigations. Therefore, this effectively simplifies the investigation process and saves time, cost and complexity.

Hence, the contributions of this research add many important aspects to the current knowledge of information hiding and detection in three perspectives: data embedding, detection of hidden messages, the evaluation of steganalysis methods, and their application as a tool for digital forensics investigation process.

1.5 Thesis Outline

This thesis comprises seven chapters. Chapter one provides an overview and the motivations of this research. It explains the research problem and the main motivations of this three-fold research with a well-defined research aim. Moreover, it states the main contributions in the area of pixel domain steganography, steganalysis and its application.

Chapter two presents steganography in detail, starting with a brief history of the discipline and defining its components then explaining the classification methods and properties of the steganographic methods. This is followed by showing the differences between steganography and other related security methods like cryptography and digital watermarking. It also goes over steganography protocols, attacks, applications, image steganography and its domains. The evaluation criteria of the steganography are another useful topic discussed in chapter two pertinent to any new method proposed.

Chapter three presents a detailed study of the steganalysis field including its definition, categories and requirements. It prioritises the main requirements of steganalysis according to importance and usability. It then explains the typical approaches of steganalysis and its three main types:

blind (or universal), semi-blind, and targeted steganalysis. As there are a number of steganalysis attacks, just like cryptanalysis, this chapter states and explains all attacks. The main objective of steganalysis is binary classification; this chapter also gives a detailed explanation about the true and false alarms with the confusion matrix. Another important aspect of steganalysis included in this chapter is evaluating its performance and showing some ways to find the best classifier. Moreover, this chapter clarifies the relation between steganalysis and digital forensics, as it could be considered as an important tool for digital investigators in relation to cybercrime, child pornography, and terrorist crimes. This chapter ends with stating some significant algorithms of steganalysis and a very useful summary about steganalysis in general.

Chapter four is about the new proposed embedding method applied in both LSB and 2LSB image steganography to improve the embedding efficiency and reduce the probability of detection by the current targeted steganalysis methods. After discussing the related works, it explains the proposed method for both LSB and 2LSB image steganography. It then shows their experimental results and compares them to the other embedding methods using the most accurate detection methods. It also analyses all the embedding methods and explains their extraction process for the proposed methods. This chapter ends with conclusions about the proposed embedding method.

Chapter five is about the new proposed targeted detection method of 2LSB steganography, EPoV. This chapter starts by explaining the 2LSB steganalysis methods and discussing the related works, then describes the concept of EPoV and applies it as a discrete and probabilistic classifier. It is experimentally shown that both classifier types are more accurate than the current targeted 2LSB steganalysis methods, and the probabilistic method gives the most accurate estimation of the embedded message size even for very low embedding rates. This chapter then ends with conclusions about the proposed detection method and its related works.

In chapter six, the application of the steganalysis is considered in relation to the digital forensics investigation process. It starts with a brief introduction and discusses the assessment of steganalysis tools, then it gives some information about this topic and explains the Stegdetect steganalysis tool, which is capable of detecting many types of steganographic algorithms. As a strongly related subject, the digital investigation process is discussed. Then, the methodology and the process of this part of the research are explained and detailed analyses of the results are given. Next, a new method of comparing the detection results is proposed. This method applies a statistical method to help the digital forensics analyst to narrow down the scope of the investigation process. The chapter ends with a conclusion that highlights the important points raised in this chapter.

Finally, chapter seven summarises the research findings and conclusions about the whole three-fold research presented in this thesis and presents an overview of the main contributions to knowledge and states the limitations of the research. It also discusses some directions for further researches that could be done in this research area.

CHAPTER 2: STEGANOGRAPHY

2.1 Introduction

With the development of the Internet, information became available online and could be accessed from everywhere, anytime. Along with that there is a rapid increase of interest in hiding information in digital media (Popa, 1998). This is simply because communication is increasingly vulnerable to eavesdropping and unwanted interventions, which could not be solved using traditional methods of cryptography (Cox, Miller, Bloom, Fridrich, & Kalker, 2008).

Digital steganography is a younger security method than cryptography, which could be defined as an art and a science of hiding secret messages in different digital media files (image, audio, video, text, etc.), so that it can be correctly received by the second party without raising the suspicion of observers (Bailey, Curran, & Condell, 2004). Steganography is considered broken when the existence of the secret message is detected. Therefore, the most important property or requirement of steganography is undetectability, which means that the message cannot be detected by any existing detection method (Fridrich, Pevný, & Kodovský, 2007).

Almost all types of digital media where there is some sort of redundancy could be used for steganography. Multimedia objects are considered excellent media for hiding secret messages because there are numerous formats with a high degree of redundancy (Chandramouli & Memon, 2001).

This chapter presents the basics of digital steganography. It starts by defining steganography and identifying its main components. Like any other systems, steganography has its own classification methods that are briefly described in this chapter. Then, it goes through the properties of digital steganography and highlights the main differences with digital watermarking and cryptography. The steganography protocols and attacks are also explained. At the end of this chapter the steganography in digital images and its evaluation criteria are explained.

2.2 Steganography Throughout History

Steganography is derived from two Greek words, *Steganos* and *Graphy,* which means “covered” (i.e. secret) “writing” (Cole & Krutz, 2003). The first written evidence about steganography dates back to nearly 440 BC, by the Greek historian Herodotus (Dunbar, 2002). Different methods of information hiding have been used throughout history. Ancient Greeks wrote messages on wood tablets covered with wax to be unseen. They also shaved and tattooed a message on the

messenger's head, which could be sent when the hair is grown back (Neil F Johnson & Sushil Jajodia, 1998). During World War I, the Germans developed microdot technology by several stages using non-suspicious cover materials like magazines (Stefan Katzenbeisser & Petitcolas, 2000). In World War II invisible inks were used to write messages between the lines of innocent letters. The open-coded messages were also used during World War II by German spies, as they did not seem to cause any suspicion (Neil F Johnson & Sushil Jajodia, 1998). For more details on the history of steganography, a large body of literature is devoted to this subject (Neil F Johnson & Sushil Jajodia, 1998; Judge, 2001; N. Provos & Honeyman, 2003).

Nowadays, with the availability of the Internet and powerful computers, steganography has developed different and clever methods of embedding in various digital media like image, video, audio and text.

2.3 Components of Steganography

The easiest way of describing the components of steganography is to consider the first invisible communication model of the prisoners' problem proposed by (Simmons, 1984). In this model, Alice and Bob are two criminals confined in two separate jail cells who want to develop a runaway plan. The warden, called Wendy, will let Alice and Bob communicate, but she monitors all their communications. Thus Alice and Bob will not be able to use encryption methods, as Wendy will stop their message exchange if she notices any suspicious communication.

Thus, they need to use a covert communication method like steganography. Hence, Alice tries to exchange a secret message *m* with Bob by embedding it into a randomly harmless message *c*, called cover object, to create a stego-object *s* that looks similar to *c* and avoids raising suspicion. A secret key could be used by the embedding process, called a stego-key *k*. Alice then sends *s* over an insecure channel to Bob, hoping that Wendy will not notice the embedded message. Then, Bob can extract the secret message *m*' from the stego-object s, since he knows the embedding method used by Alice and has access to the key used in embedding process. The extraction process should be possible without referring to the original cover (Stefan Katzenbeisser & Petitcolas, 2000). This is illustrated in Figure 2.1.

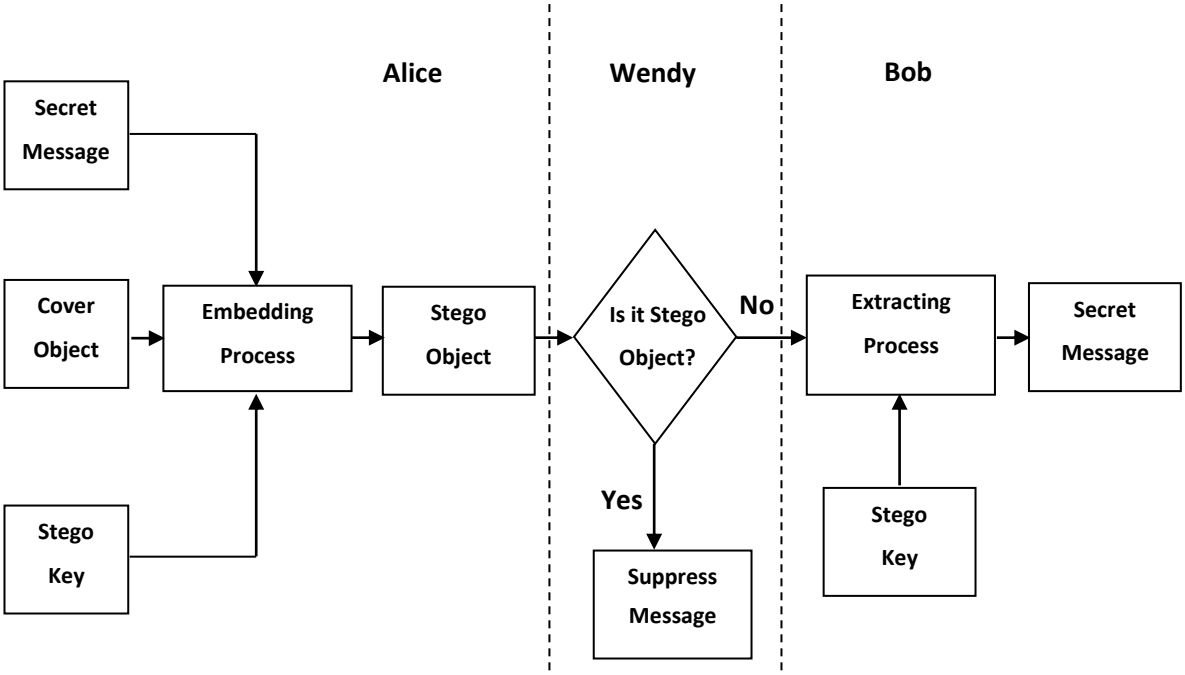


Figure 2.1: Schematic description of Steganography Framework

The warden, Wendy, could test the stego object to see if there is any secret message hidden by Alice. If she did not find any sign of hidden data, she will let the stego object pass to Bob (passive warden). However, Wendy could be an active warden and modify everything exchanged between them irrespective of whether there is any indication of having hidden information or not (Chandramouli, 2002).

The followings are very short definitions of some naming conventions of steganography framework notations adopted after the first Information Hiding Workshop (Embedded, 1996).

2.3.1 Cover Object

The cover object could be any type of digital media with some redundancy in their representation format, like text, image, audio, video etc. Each cover object has a specific embedding capacity depending on the media type and the embedding method. The steganographer is free to choose any cover object, unlike digital watermarking, in which the selection of the cover object is restricted.

2.3.2 Secret Data

Secret data could be any stream of binary representation that needs to be transmitted over an insecure channel without raising suspicion. Hiding more data in general increases the probability of detection, so choosing an appropriate cover is crucial.

2.3.3 Embedding Process

The embedding process usually has three inputs; cover object, secret data, and an optional stego key. It uses a particular method, for example LSB replacement, to embed the secret data into the cover object and create the stego object as an output.

2.3.4 Stego Object

This is the modified version of the cover object after embedding the secret data, which should look similar to the cover object. The stego object should at least maintain the imperceptibility property, explained in section 2.5.2, and not be degraded by the embedding process.

2.3.5 Stego Key

The stego key is a secret key used in the embedding process to make the secret data computationally infeasible to extract by the extraction process without having access to that secret key. It can be a number generated via a pseudo-random number generator (Chandramouli, Kharrazi, & Memon, 2004), or just a password for decoding the embedding location. The secret data may also be encrypted before embedding; in this case the recipient needs two keys to get the secret data, one for extracting and the other for decrypting the secret data (Cox et al., 2008).

2.3.6 Extraction Process

This is an opposite function of the embedding process; it takes the stego object and an optional stego key as an input and extracts the secret message as an output. This process could be achieved without referring to the original cover. However, it is possible to extract the hidden message by comparing both stego and cover object. Both cases are explained in sections 2.4.3.1 and 2.4.3.2.

2.4 Classification Methods of Steganography

There are many methods of classifying steganographic systems. These classification methods are very useful for studying and developing embedding and detection methods, as each type of steganography has its own properties and attributes. The main classification approaches are based on the cover type, hiding method and extraction method (Cole & Krutz, 2003; Cox et al., 2008; Kipper, 2004).

2.4.1 Based on the Cover Type

As there are many different types of digital media that can be used as cover files for embedding secret data (Cole & Krutz, 2003), steganography methods could be classified based on the utilised

cover type; for example, image steganography is a method of embedding secret data into digital images.

The cover-type method of classification is very important as different media files have different properties, which could be exploited in such a way that possibly gives minimum probability of detection. It is also useful for steganalysers, as they develop detection methods by finding abnormality in properties of a specific digital media.

2.4.2 Based on Hiding Method

Another approach of classifying steganographic systems is the method of embedding. In general there are four different ways to hide secret data in cover files; insertion based, substitution based, generation based, and cover lookup based (Cole & Krutz, 2003; Cox et al., 2008; Kipper, 2004).

2.4.2.1 Insertion Based

Since there are some areas in cover files that are usually ignored by the application that reads the file, insertion based steganography inserts the secret data into these areas of the cover file. Hence, this method keeps the readable areas unchanged, which is an advantage of this embedding method. Thus a huge amount of secret data could be inserted to any cover file without limiting the embedding capacity. However, the size of the stego file would be much larger than the size of the cover file by adding the secret data to the cover file without removing any bit of it (Cole & Krutz, 2003). This difference in size could sometimes make the stego file suspicious, without looking at the content.

A good example of such a method is inserting a secret message in the area between end-text and begin-text markers of a Word document. This is because the Microsoft Word application is configured to ignore anything written in this area and the secret data will not be displayed at the time of viewing the document file (Cole & Krutz, 2003).

2.4.2.2 Substitution Based

Substitution based or cover modulation is the most common and the most advanced method of steganography (Cox et al., 2008). This method, unlike the insertion based method, looks for some insignificant areas in cover files and replaces them with the secret data (Cole & Krutz, 2003). Therefore, the quality of the stego file would be degraded by the embedding process compared to the quality of the cover file. Also, according to the insignificant amount of information in the cover file, there is a limitation on the size of the secret data that could be hidden. However, both stego and cover files will have the same size, because they only modify some insignificant parts of the cover file without inserting any additional data.

The embedding process should rely on some methods for selecting the location of the change. In general, there are three selection rules to follow in order to control the selection of the location of change; sequential, random and adaptive (Cox et al., 2008).

A sequential selection rule modifies the cover object elements individually by embedding the secret message bits in a sequential way. For example, it is possible to embed the secret message by starting from the top-left corner of the image to the bottom-right corner in a row-wise manner. This selection rule (sequential) is very easy to implement, but has very low security against detection methods.

A pseudo-random selection rule modifies the cover object by embedding the secret message bits into a pseudo-randomly chosen subset of it, possibly by using a secret key as a pseudo-random number generator (PRNG). This type of selection rule gives a higher level of security than sequential rule.

An adaptive selection rule modifies the cover object by embedding the secret message bits in selected locations based on the characteristics of the cover object (e.g. choosing noisy and high textured areas of the image, which are less detectable than smooth areas for hiding data). Adaptive selection rule gives higher security than sequential and pseudo-random selection rules in terms of detection.

2.4.2.3 Generation Based

Generation based method steganography, or cover synthesis, is different from the two previously mentioned methods. In this method, no cover files are used by the embedding process; rather it uses the secret data to generate a suitable stego file (Cole & Krutz, 2003; Cox et al., 2008). Therefore it cannot be detected by the detection methods that rely on comparing stego with the cover file, as the cover file does not exist.

However, this method has a limited number of stego files that could be generated; also, it may generate impractical files like images with lots of random shapes and colours that make no sense, or generating a text with no meaning (Cole & Krutz, 2003). In other words, the generated stego file might look suspicious to human perception, which is less predictable than automated detection methods.

Mimic functions is a good example of steganography by synthesis (Wayner, 1992). It encodes a short message into a proper spam document. Another example of synthesizing the cover work is data masking (Radhakrishnan, Kharrazi, & Memon, 2005), whereby a secret message is shaped into a stego cover whose statistical properties are like a normal cover file, such as music.

2.4.2.4 Cover Lookup Based

This method looks for a pre-existing cover file so as the secret data embedding does not need to modify anything in the cover file. It assumes that it can find the appropriate cover file that already holds the desired secret data. Therefore, as the size of secret data increases, this solution becomes unusable very quickly. For example, if sending 20-bits of secret data requires a million cover files, then 30-bits requires a billion cover files (Cox et al., 2008).

2.4.3 Based on Extraction Function

The steganographic systems can also be classified as blind and non-blind (or informed) schemes according to whether the original media is used or not respectively by the extraction process. However, this type of classification is missing in steganographic literature (Cox et al., 2008).

2.4.3.1 Blind Steganographic Scheme

Steganographers usually assume that the cover medium is unnecessary for the extraction process by the recipient in their proposed steganographic methods. Hence, the extraction process does not need the original cover media and can get the hidden information back from the stego media only (Cox et al., 2008; L. M. Marvel, Boncelet, & Retter, 1999). Consequently, it enables Alice to use any cover media, even if it is not accessible by Bob.

2.4.3.2 Non-blind Steganographic Scheme

In non-blind or informed steganographic scheme, the original cover medium is considered necessary by the extraction process and the retrieval of hidden information would be impossible without it. Unfortunately, this steganographic scheme is generally neglected by steganographers, despite its potential practical utility. For example, if Alice and Bob both agreed to use the same set of images, the informed extraction process would help the embedding process to embed the hidden information in a less strong manner. As a result, the probability of detection by the attacker would be smaller than blind embedding methods (Cox et al., 2008).

2.5 Properties of Steganography

As steganography belongs to a wider field called information hiding, all properties of information hiding could be considered for both steganography and digital watermarking. However, steganography defines and prioritises these properties slightly different from watermarking, as explained below.

2.5.1 Undetectability

The main purpose of steganography is to hide data in such a way that makes the existence of the hidden message secret. Thus, undetectability is the most important property of any steganographic system, which means that the existence of the secret data cannot be noticed by the use of statistical approaches of detection. If someone could easily recognise the stego media, then using such a steganographic method makes no sense (Cole & Krutz, 2003). There are a number of factors that directly affect the undetectability, for example the choice of the cover media, the method of embedding and the number of changes introduced to the cover media (Fridrich, Lisoněk, & Soukal, 2007).

However, there is no steganographic method that can embed data into a certain media file without leaving some artefacts. Hence, the lower probability of detecting these artefacts denotes a better steganography method. That is why developing a new steganographic method is not enough if it does not confer less probability of detection by current steganalysis methods.

2.5.2 Imperceptibility

Imperceptibility is another property of steganographic systems, which means that the stego media should not have any noticeable artefacts after embedding secret data (B. Li, He, Huang, & Shi, 2011). Hence, most steganographic methods utilise the limitation of the Human Visual System (HVS) or Human Auditory System (HAS) in their embedding process (VenkatramanS, Ajith, & Paprzycki, 2004). For example, the stego image should look like an innocuous image by HVS.

There are multiple evaluation criteria for imperceptibility, which could be considered according to the type of steganographic method and/or the type of the used cover file for data hiding. For example, the file size could be an indicator of having hidden data in text files or insertion based steganography, whereas image quality could be an indicator for hidden data in substitution based image steganography. However, currently most steganographic methods have a high level of imperceptibility, but they suffer from statistical detection.

2.5.3 Security

The term “security” in steganography literature is used as an equivalent word to “undetectability”; thus a steganography method is considered secure when it is statistically undetectable (Cox et al., 2008). Most current steganographic methods are considering passive wardens, whereas active wardens, as discussed by (Craver, 1998), have been considered much less in the literature.

2.5.4 Capacity

There are two different types of capacity in relation to the field of steganography: the embedding capacity and the steganographic capacity (Cox et al., 2008). Embedding capacity is the maximum number of bits that can be embedded in a certain media file. For example, the embedding capacity for a grey-scale image with LSB replacement would be equal to the total number of pixels in the image. The steganographic capacity is different from embedding capacity and is not easy to determine even for a very simple embedding method. It could be defined as the maximum number of bits that can be embedded in a certain media file with such a probability of detection that can be neglected by the attacker.

2.5.5 Robustness

In general, there are two factors that affect the robustness in steganography. First, is the undetectability, which is explained in previous sections. Second is the ability to defeat the active attack, which is more important for digital watermarking (Wang & Wang, 2004), which means that the secret message should be recovered by the second party even if the cover media faced some data processing (Cole & Krutz, 2003). A steganography method could be considered as robust if both the detection and the destruction of the hidden data are hard.

However, as claimed by (Cox et al., 2008), defeating the active attack is rarely considered for steganography because it is assumed that the stego object will be sent over the Internet, whereby there would not be any degradation and the second party would receive exactly what the first party sent.

2.5.6 Conflicts Between Requirements

The main goal of improving any steganographic method is to enhance its requirements in terms of undetectability, imperceptibility and capacity (Chang, Lin, & Wang, 2006). However, enhancing a certain requirement may negatively affect others. For example, the undetectability and capacity cannot be maximised at the same time. The amount of artefacts produced in the cover media by the embedding process is directly affected by the amount of hidden data (L. Marvel, Boncelet, & Retter, 1998).

As a result, there should be a trade-off among these requirements. Steganographic systems must achieve a high imperceptibility and a high capacity, but they are not necessarily robust in terms of defeating active attacks. However, the robustness would be the highest priority requirement for digital watermarking schemes (L. M. Marvel et al., 1999).

The embedding domain also satisfies these requirements in different levels. Data embedding in the frequency domain is more robust than in the spatial domain. However, spatial domain data embedding has a higher capacity than frequency domain schemes (Y.-H. Yu, Chang, & Hu, 2005). Therefore, most digital watermarking schemes are developed based on frequency domain embedding schemes. Moreover, most steganographic methods are proposed to enhance the security and the capacity of steganography methods (Chen & Lee, 2003; Hsien-Wen & Chin-Chen, 2004; Y. K. Lee & Chen, 2000; Qingzhong, Chen, & Dongsheng, 2006; Rufeng, Xinggang, Xiangwei, & Xiaohui, 2004).

It is possible to find a clash between undetectability and the imperceptibility requirements as well. For instance, the stego media could be statistically undetectable using cover generation methods, while visually suspicious due to unrealistic media, such as random shapes and colours in images or a text that makes no sense (Cole & Krutz, 2003).

2.6 Steganography and Cryptography

Steganography and cryptography are intended to accomplish different goals; steganography keeps the existence of the message secret, whereas cryptography keeps the content of the message secret (Lou & Liu, 2002). Therefore, even though the message is encrypted, the existence is still a major weakness of cryptography methods. Thus steganography intended to supplement cryptography, rather than replacing it, and putting both methods together will add another layer of security by making a scrambled message hidden.

Although both steganography and cryptography systems offer secret communications, they have dissimilar breaking definitions. A steganography system is considered broken if the eavesdropper could detect the existence of the secret message. However, a cryptography system is considered broken if the eavesdropper could read the content of the secret message (Zöllner et al., 1998).

2.7 Steganography and Watermarking

Both steganography and watermarking are related to a broader subject known as information hiding. They both share some properties like imperceptibility, robustness, capacity and security. However, they prioritise these properties differently; for example, imperceptibility is the most important requirement for steganography (Morkel, Eloff, & Olivier, 2005), while for watermarking the robustness has higher priority (Boato, Conotter, De Natale, & Fontanari, 2009).

However, in some cases the imperceptibility is not an issue in steganography, for example when changing the colour of a certain object in the image. This change may still be undetectable, since the third party has no access to the original image (Cox et al., 2008).

Steganography and watermarking are intended to achieve different functions; steganography is used to protect the secret message using another digital object to provide hidden communication, whereas watermarking is intended to protect the cover object by embedding a special watermark for copyright protection. Watermarking and fingerprinting are very close; they both mark objects in the same way, except in watermarking all objects have the same marking embedded for copyright protection, whereas in fingerprinting objects are marked separately for each customer to prove the ownership (Anderson & Petitcolas, 1998).

Another difference is the cover object itself. In the case of steganography, Alice is free to select which cover object to use; thus she can avoid cover objects in which it is difficult to conceal a message. Conversely, in digital watermarking the cover object is specific and cannot be avoided (Cox et al., 2008).

2.8 Steganography Protocols

Generally there are three types of protocols: pure steganography, secret key steganography, and public key steganography.

2.8.1 Pure Steganography

Pure steganography is a class of steganography system whereby there is no prior information shared by two communication parties (Stefan Katzenbeisser & Petitcolas, 2000). In this case, both Alice and Bob must have access to embedding and extraction functions, and these functions should not be known by the third party. In practice, pure steganography is not very secure because it is not consistent with Kerckhoff's principle, which assumes that the embedding algorithm is known to Wendy (Francois Cayre, Fontaine, & Furon, 2005).

2.8.2 Secret Key Steganography

According to Kerckhoff's principle, as Wendy has access to the extraction method, she is able to extract the hidden information from every stego media exchanged between Alice and Bob. Therefore, the security of the hidden data should depend on some secret information exchanged by Alice and Bob, called the stego-key. Without having this key, nobody should be able to obtain the secret information from the stego media (Francois Cayre et al., 2005). However, the additional transmission of the secret key is inconsistent with the main purpose of steganography, which is

invisible communication; it could be assumed that Alice and Bob had agreed on a stego-key before detention.

2.8.3 Public Key Steganography

The public key steganography uses two keys: the public key and the private key. The public key is stored in a public database and used by the embedding algorithm, whereas the private key is used by the extraction algorithm to recover the secret message. Thus the public key steganography can be built using public cryptosystem, in which Alice and Bob do not need to exchange the secret key. Again, it is assumed that Alice and Bob have exchanged their public key before imprisonment (Stefan Katzenbeisser & Petitcolas, 2000).

As it is considered that the embedding method is known to Wendy, she can try to extract the hidden message in the stego media. However, in this case, she will not be able to recognise the secret message because it should look like a random string of bits due to encryption.

2.9 Attacks on Steganography

As steganography system is mainly presented in the form of prisoners' problem (Simmons, 1984), so the warden would be the attacker and she has the possibility to attack the steganographic communication between Alice and Bob in three different ways: passive, active and malicious warden.

2.9.1 Passive Warden

The passive warden monitors the communications between both parties (Alice and Bob) and tests for the existence of hidden information. If a hidden message is detected, then she blocks the transmission. Otherwise, if it is not detected, then she lets the communications to be forwarded (Cox et al., 2008).

2.9.2 Active Warden

The active warden, unlike the passive warden, is capable of modifying communication between Alice and Bob. Thus, the communication could be altered even if the hidden message is not detected in order to destroy any undetectable hidden message (Chandramouli et al., 2004).

2.9.3 Malicious Warden

The malicious warden may have further capabilities, like sending false messages to Alice and Bob. In this case, in addition to detecting the hidden message, the warden knows the steganographic algorithm used by Alice and Bob and also any keys related to their embedding algorithm

(steganographic and cryptographic keys) (Chandramouli et al., 2004). Hence, the complexity of the steganography method and the amount of prior knowledge will indicate the difficulty of the warden's task (Chandramouli, 2002; N. Provos & Honeyman, 2003).

2.10 Applications of Steganography

Steganographic methods could be used by any two parties that might wish to protect the secrecy of their communication. Also, there are numerous reasons why people or agents want their communication to be secret. For example, they could be two lovers who wish to hide their relationship, or forbidden political organisations that want to communicate among themselves, or even criminals who want to organise a crime or a terrorist operation (Cox et al., 2008).

There are other specific uses of steganography methods like controlling copyright protection, improving the robustness of image search engines, and smart identity cards (Jain & Uludag, 2002). Moreover, steganography methods could be used for embedding checksums and error correction codes (Bender et al., 2000; Chang, Hu, & Lu, 2006). Another application of steganography methods is to maintain the link between image data and the patients' information, whereby the separation is considered necessary for confidentiality purposes, by embedding patients' information into the image (Stephan Katzenbeisser & Petitolas, 2000). Some other methods of steganography have been discussed in relation to patient records and data concealment in digital images (Anand & Niranjan, 1998; Shaou-Gang, Chin-Ming, Yuh-Show, & Hui-Mei, 2000; Yue, Chang-Tsun, & Chia-Hung, 2007).

Secret communication may be used in the business sector as well, and in the modern economic climate the security of corporations is no less important than the security of countries, all large organisations must protect their online information using steganography and other security methods.

2.11 Steganography in Digital Images

Many methods of secret communication have been developed in the last few decades, among which image steganography is one of the major areas (Chandramouli et al., 2004; N. Johnson & S. Jajodia, 1998; B. Li et al., 2011; N. Provos & Honeyman, 2003; Wang & Wang, 2004). This is because there are millions of images on the web in which anyone can embed their own messages for the purposes of covert communication (Wayner, 2002). Also, digital images have a high degree of redundancy in representation, and small changes to digital images cannot be observed by HVS.

Moreover, they can easily be used as cover media for data embedding without raising suspicion due to omnipresence on the web (Artz, 2001; Liu & Liao, 2008). Therefore, digital images are the most widely used cover media for steganography.

Almost all steganographic systems exploit the characteristics of human visual system in their embedding methods (Artz, 2001; Chang, Chen, & Chung, 2002; Chang & Tseng, 2004). Hence, steganographers are interested more in noisy and edge regions in the image than smooth areas, as the HVS is less sensitive to the degradation in noisy and edge regions (Zeng, Lin, & Yu, 2006).

In spite of having some progress of image steganography in binary images (Liang, Wang, & Zhang, 2007; Min, Tang, & Lin, 2000) and 3-D images (F. Cayre & Macq, 2003), researchers mostly focus on hiding data in grey-scale and colour images. Although the luminance component of a colour image is equivalent to a grey-scale image, some experts consider grey-scale images as best cover for steganography (Aura, 1996; Fridrich, Goljan, & Du, 2001b). This is because the embedding process changes the correlation between colour components, which makes the trace of embedding easier to reveal.

In general there are two main types of image steganography, spatial domain and transform domain, as explained in brief in the following sections. However, there are some other types of image steganography that are less common. For example, appending the secret messages to the end of file (EOF) tag of the JPG image file is very simple, and it will be ignored by image viewer applications for display. This type of data embedding is very simple, does not affect the image quality, does not change the image histogram, and it is imperceptible when opened by image viewer applications. However, if the stego image is opened by other applications like Notepad, the message will be shown as the Notepad is not configured to deal with EOF tag of the JPG file. Another example is appending hidden data to the image's extended file information (EXIF), which is used by manufacturers of digital cameras to store information like the camera's make and model, the time of capturing the photo and its resolution etc. As claimed by (Alvarez, 2004), the EXIF information could help verifying the authenticity of a picture in an investigation process in relation to child pornography.

The most important thing is that appending hidden data into metadata tags of the image file cannot resist any kind of editing or attacks (Cheddad, Condell, Curran, & Mc Kevitt, 2010). Also, it could be noticed from the size of the file, especially when the hidden data is relatively large (Cole & Krutz, 2003).

2.11.1 Spatial Domain Image Steganography

The general idea of image steganography in spatial domain is to directly modify the value of image pixels in order to embed the secret message. The simplest and the most common spatial domain image steganography is LSB replacement, which directly replaces the LSB of the selected (sequentially or randomly) image pixel values with one bit of the secret message (Mielikainen, 2006). This type of embedding, LSB replacement, is the most widely used embedding method in special domain, which has a low computational complexity and a high embedding capacity (L. Yu, Zhao, Ni, & Li, 2010). Another reason behind the popularity of LSB steganography is that it has a large embedding capacity without introducing noticeable distortions (Bender, Gruhl, Morimoto, & Lu, 1996). However, it produces pairs of values (PoV) in the stego image histogram and gives steganalysers an opportunity to successfully detect the hidden content (Westfeld & Pfitzmann, 2000).

In order to avoid the statistical attack on the resultant pair of values, LSB matching (X. Li, Yang, Cheng, & Zeng, 2009; Mielikainen, 2006; Sharp, 2001) is developed. LSB matching is a modified version of LSB replacement, which randomly adds or subtracts 1 to the pixel value with mismatched LSB instead of simply flipping the LSB value from 0 to 1 or vice versa. This type of embedding, ±1 embedding, causes the distortion to the cover image as an additive independent identically distributed (i.i.d.) noise, which may lead to a successful steganalysis (Giacomo Cancelli, Doërr, Cox, & Barni, 2008; Harmsen & Pearlman, 2003; A. D. Ker, 2005b; J. Zhang, Cox, & Doërr, 2007).

Although there is a difference in embedding function between LSB replacement and LSB matching steganography, their extraction methods are the same. The secret message could be extracted directly from the LSBs of the image pixel values. Another noise adding steganography method, stochastic modulation, is presented by Fridrich (Fridrich & Goljan, 2003). It embeds the secret message by adding a weak noise signal with a specified arbitrary probabilistic distribution. This embedding method enables the user to mask the embedding distortion as noise generated by a particular image acquisition device.

There are a number of image steganography tools, some with open source codes, that use the spatial domain embedding methods of steganography in digital images; more details could be found in a survey article by (Hayati, Potdar, & Chang, 2007).

The extensions of LSB steganography and steganalysis received less attention by researchers prior to the last decade. However, nowadays there are many steganography tools that allow the use of

more than one LSB for data embedding; SilentEye (Chorein, 2008) is a very good example of such a steganographic tool.

In spite of extreme easiness of implementing data hiding in multiple bit-planes, the non-adaptive multiple bit-plane steganography may negatively affect the imperceptibility and the quality of the stego image, because the high bit-planes may involve in modification (B. Li et al., 2011). However, the local property of the image pixel could be considered for developing an adaptive multiple bit-plane data embedding. The bit-plane complexity segmentation (BPCS) developed by (Kawaguchi & Eason, 1999) is a good example of considering the local property in each block. The BPCS uses the cover image as vessel data and embeds the secret message in its bit-planes. It also utilises the human vision system and replaces the "noise-like" regions with the secret data.

Also, ±k steganography (Fridrich, Soukal, & Goljan, 2005), of which ±1 embedding is a special case, is another extension to LSB steganography. Instead of simply replacing the k bit-planes of the pixel value with k bits of the secret message, the ±k embedding increases or decreases the pixel value by k to match the k-LSBs of the pixel value with k bits of the secret message. The distortion of non-adaptive ±k embedding could be represented as an additive independent identically distributed (i.i.d.) noise signal with the following probability mass function.

$$P_{+k} = \frac{p}{4} \quad , \quad P_0 = 1 - \frac{p}{2} \quad , \quad P_{-k} = \frac{p}{4}$$

Where, p is the embedding rate in bits per pixel.

Therefore, 2LSB steganography would be a very good extension method for LSB steganography, because it is still imperceptible, easy to implement, and even has a higher capacity. Moreover, it introduces complex modifications of pixel values, which makes existence of the secret message very hard to detect (Niu, Sun, Qin, & Xia, 2009).

2.11.2 Transform Domain Image Steganography

The embedding of secret data in transform domain steganography is done by modulating the coefficients in transform domain, such as in discrete cosine transform (DCT), discrete Fourier transform (DFT), and discrete wavelet transform (DWT) (Amin, Salleh, Ibrahim, Katmin, & Shamsuddin, 2003). Transformed domain data embedding is commonly used to perform high capacity embedding in steganography and to obtain a robust data embedding in digital watermarking (Wang & Wang, 2004). In general, the insignificant areas of the transformation coefficients would be used to have a high capacity and imperceptibility, whereas significant areas are used to get robust data embedding against active attack.

The transform data hiding technique could be integrated with compression algorithms like JPEG (Joint Photographic Experts Group), whereby the secret message is embedded by modifying the DCT coefficients of the transformed cover media, and then it takes the inverse DCT to create the modified image (stego). This method of data hiding is very attractive due to the massive availability of JPEG images on the web (Wang & Wang, 2004).

Another approach is spread spectrum steganography (L. M. Marvel et al., 1999; Smith & Comiskey, 1996), which could be used in both domains. It embeds the secret data by spreading it throughout the cover image by modulating a carrier function (a common choice is Gaussian random vector) to make it less detectable. There are three common spectrum spreading schemes (Smith & Comiskey, 1996): direct sequence, frequency hopping and chirp.

2.11.3 Adaptive and Non-Adaptive Image Steganography

Non-adaptive image steganographic techniques modify the cover image for message embedding without considering its features (content). For example, LSB replacement and LSB matching with sequential or random selection of pixels modify the cover image according to the secret message and the key of random selection of pixels without taking the cover image properties into account. Adaptive image steganography techniques modify the cover image in correlation with its features(Fridrich & Du, 2000). In other words, the selection of pixel positions for embedding is adaptive depending on the content of the cover image. The bit-plane complexity segmentation (BPCS) proposed by (Kawaguchi & Eason, 1999) is an early typical method of adaptive steganography.

As the adaptive steganographic schemes embed data in specific regions (such as edges), the steganographic capacity of this method is highly dependent on the cover image used for embedding, it is expected to have a lower embedding rate than non-adaptive schemes. However, steganographers have to pay this price in order to have a better security or less detectable stego image.

2.12 Steganography Evaluation Criteria

Currently there is no standard measurement available to evaluate the performance or effectiveness of steganographic systems. However, it is very important to have such an evaluation scheme. In order to decide which steganographic system or method has superiority over another, there are some evaluation criteria that can be considered for evaluating any steganographic system in terms of its most important requirements: security, capacity and imperceptibility (B. Li

et al., 2011). These evaluation criteria could also be used to improve the current embedding techniques.

2.12.1 Evaluation of Security

The security or undetectability is the main requirement of any steganographic system; the optimum scenario is to make the probability of detection no more than a random guess (Fridrich, Lisoněk, et al., 2007). It also could be tested practically by looking at the specific current steganalysis methods to estimate the probability of detection. However, it is almost impossible to have a completely secure steganographic method as they embed the secret messages in the actual component of the cover object and thus modify it. Instead, steganographers evaluate the security of the embedding method by its relative probability of detection.

2.12.2 Evaluation of Capacity

Capacity, for evaluation purposes, means the steganographic capacity and not the embedding capacity. The steganographic capacity is the maximum number of bits that can undetectably be hidden (Cox et al., 2008), which is not easy to find even for a very simple embedding method. As there is a trade-off between the capacity and both undetectability and imperceptibility, a significant contribution is achieved if a certain steganography method could maintain the same steganographic capacity with higher imperceptibility (N.-I. Wu & Hwang, 2007). Also, it is good to have a higher steganographic capacity and maintain an acceptable level of imperceptibility. Another significant contribution could be considered if a certain steganography method could maintain the same probability of detection with a higher capacity.

2.12.3 Evaluation of Imperceptibility

In signal processing systems, two types of evaluating the imperceptibility can be distinguished; fidelity and quality. The perceptual similarity between signals before and after processing is known as fidelity, whereas the absolute measure of the goodness of a signal is called quality (Almohammad, 2010). For steganography in digital images, the fidelity is the perceptual similarity between the original cover and the stego image. Hence, both images are considered important for fidelity evaluation.

There are two main methods of measuring the quality of images: objective (automated) or subjective (human based) (Stoica, Vertan, & Fernandez-Maloigne, 2003). The objective methods consider the physical aspects of images using mathematical calculations or model. Typical examples of objective methods are Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR),

Structural Similarity (SSIM), Multi-Scale Structural Similarity (MSSIM), Visual Information Fidelity (VIF), and Visual Signal to Noise Ratio (VSNR).

The subjective methods are perceptual based assessment of the image quality. According to the availability of the original (unmodified) image, the subjective image quality metrics can be classified into three main categories (Zhou, Bovik, Sheikh, & Simoncelli, 2004): full-reference, no-reference, and reduced-reference. The full-reference assumes that both the original and test images are available, whereas no-reference assumes that the original image is not available. However, reduced-reference assumes that the original image is partially available (i.e. only some information or features) (Ponomarenko et al., 2008).

Moreover, to avoid attracting attention by the third party, suspension, and also detection, steganography systems should use very good quality images (Cox et al., 2008). Therefore, evaluating the quality of stego images is a very important indicator of the performance of any image-based steganography method (N.-I. Wu & Hwang, 2007).

Despite of being fidelity metrics by definition, PSNR and MSE are generally known as quality measures used to measure the amount of distortion added to an image in the form of perceptual distance metrics. Therefore, the fidelity is defined as the perceptual quality of stego images, and both PSNR and MSE characterise the imperceptibility of the secret message (Cox et al., 2008).

2.13 Summary

The key concerns and considerations of steganographers have been explained in this chapter. Any steganographic scheme is considered broken when the existence of the hidden message is detected, thus the statistical undetectability of the embedded data is the most important property for any steganographic system. Regarding the embedding methods, LSB embedding is the most common steganographic method in spatial domain because it has a reasonable capacity, is easy to implement, and visually imperceptible.

The steganalysis methods could be utilised to improve the security of the steganographic methods. This could be achieved by reducing those artefacts that cause the detection and defeating or reducing the probability of detection. The average number of random bits per one embedding change is called embedding efficiency. In general, improving the embedding efficiency will reduce the probability of detection by producing fewer changes in the stego image. The embedding efficiency of a typical LSB embedding is two random bits per one embedding change.

Considering all classification methods, the most widely used steganographic method is LSB embedding in images, which is a blind substitution based method of embedding.

There is a difference between embedding capacity and the steganographic capacity. The embedding capacity is the maximum number of bits that can be embedded in a certain file, whereas the steganographic capacity is the maximum number of bits that can be embedded in a certain media file with an insignificant probability of detection.

There are a number of evaluation criteria for steganographic systems, including security, capacity and imperceptibility. However, the security of the steganographic system could be maintained when the probability of detection is low, because nowadays almost all steganographic methods are merged with some kind of data encryption. Moreover, the modern steganographic schemes have a high level of imperceptibility, but the capacity (steganographic capacity) is still an issue that could be further improved.

CHAPTER 3: STEGANALYSIS

3.1 Introduction

To illustrate steganalysis, we can imagine the scenario of Simon's prisoner problem (Simmons, 1984). In this scenario, Alice and Bob are imprisoned in a jail and are monitored by a warden, called Wendy. Alice and Bob want to discuss an escape plan and they can do so only if they could make their communication hidden by using a steganographic method for hiding their secret message exchanges. Now, as discussed by (Chandramouli et al., 2004), steganalysis can be defined as a set of methods that help Wendy to detect the existence of a secret message inside the stego-object without requiring any knowledge of the secret key, and in some cases even the algorithm of the embedding process. The absence of prior knowledge about the embedding process makes the steganalysis process in general very complex and challenging. In this setting, Wendy (the active warden) can sometimes actively stop and modify any message she feels uncomfortable with, and in other cases she is only supposed to pass messages between the two communicating parties (passive warden).

The detection of a hidden message in digital media is usually represented as a classification problem. In other words, steganalysis algorithms receive digital objects as an input and classify them into either 'Clean' or 'Stego' objects. Therefore, some other classification tools like pattern recognition and machine learning can be used for steganalysis as well. Since, the classification of digital media into clean and stego objects only has two classes, the term detection is more commonly used than classification for steganalysis methods (Cox et al., 2008).

As steganography hides information in plain sight, it becomes almost impossible for law enforcement to detect the existence of hidden content in digital images through visual examination (Craiger, Pollitt, & Swauger, 2005). That is why steganalysis tools have recently became very important and essential to law enforcement, especially in cybercrime and copyright protection (Fridrich & Goljan, 2002). Also, there is a claim that steganography has been used by terrorists and child pornographers; however there is no definitive evidence for that (Cox et al., 2008).

In this chapter, the main aspects of steganalysis are explained, starting by categorising the steganalysis methods and showing the levels of their requirements. Some typical steganalysis approaches are then stated with their main types according to the range of detection. The different types of steganalysis attacks are also listed and explained in this chapter. As steganalysis

could be modelled as a classification problem, a separate section is specified in this chapter to deal with steganalysis as a binary classifier. Moreover, it shows the performance evaluation of steganalysis and gives very good information about its relation to digital forensics. Finally, some significant steganalysis algorithms of LSB embedding and their detection principles are explained, and this chapter ends with a summary in the area of steganalysis.

3.2 Steganalysis Categories

The main objective of developing any steganalysis technique is to classify the analysed digital media into clean and stego objects. However, in certain cases like digital forensics investigations, there might be further requirements like recovering the secret message. Hence, the requirements of steganalysis may vary from one application to another. Therefore steganalysis techniques could be divided into two main categories; passive and active staganalysis.

3.2.1 Passive Steganalysis

This is the most common type of steganalysis technique; it can only detect the existence of the secret message, without giving any information about the type of steganography used or the attributes of the secret message itself (Chandramouli, 2002). The majority of current steganalytic methods can be classified as passive steganalysis, and the general idea is the utilisation of first-order or high-order statistics according to the steganography technique. When the steganography technique is unknown, there are two other approaches to follow; considering the image characteristics as an a priori model, or using a large image database as training set (Chandramouli, 2003). Irrespective of how good the detection method is, a very general steganalysis method may not perform very well on a specific steganography method. For that reason, the choice of the right steganalysis algorithm itself is an open research problem.

3.2.2 Active Steganalysis

This type of steganalysis techniques is less common, and is sometimes referred to as forensic steganalysis (Cox et al., 2008). Apart from detecting the presence of the secret message, it tries to extract an approximate version of the secret message or at least extract some attributes of it, like the message length, the location of hidden data or the secret key. However, extracting the secret message is much more complicated than mere detection (Chandramouli, 2003). Hence, the difference between active steganalysis and an active warden could be noted. Active steganalysis tries to extract the secret message without destroying the stego-object, whereas active warden tries to modify the sego-object, aiming to destroy the secret message.

3.3 Steganalysis Requirements

Although the main requirement of any steganalysis approach is the detection of hidden contents, other requirements may become important according to the purpose of applying a particular steganaysis method. In general, there are different levels of requirements which could be listed as detection only, and further requirements like indicating the steganography type to retrieve the embedded secret data.

3.3.1 Detection or Classification Only

The scope of steganalysis is usually focused on detecting the secret message and not extracting it, which is reasonable enough as the main goal of steganography is to conceal the secret message (Stefan Katzenbeisser & Petitcolas, 2000). This type of steganalysis technique is known as passive steganalysis (Chandramouli, 2002). In this case, detecting the existence of hidden messages in digital media is considered as a classification problem. Hence, it is required from the steganalysis method to distinguish between clean and stego media only, without any considerations about the retrieval of the embedded message.

3.3.2 Further Requirements

After the first level of requirement, which is detection, there might be other requirements like recovering some attributes of the hidden message (e.g. estimating the message length and the location of embedding), revealing the class of steganography algorithm (Cox et al., 2008), or even retrieving the embedded message from the stego object. This type of steganalysis technique is also known as active steganalysis (Chandramouli, 2002). As mentioned previously, recovering the secret message is much more complicated than detection only, requiring the knowledge about the embedding algorithm, estimation of message length, and probably both encryption key and algorithm (Fridrich, Goljan, Hogea, & Soukal, 2003).

3.4 Typical Steganalysis Approaches

Steganography hides the secret message in different media types in such a way that makes the existence of the hidden message secret. However, due to the modifications in the carrier media, there should be some artefacts that could indicate the existence of the embedding process.

There are different approaches that could be applied to observe the artefacts caused by the embedding process, the basic approaches are visual, structural and statistical steganalysis.

3.4.1 Visual Steganalysis

Visual attacks (and aural for audio files) are the simplest approach of steganalysis. The visual attack removes significant parts from the media file and utilises human senses for examination (i.e. seeing and hearing), as human senses are capable of complex analysis that can outperform the power of computers in many ways (Wayner, 2002).

The most common visual attack on images is done by displaying the least significant bits of an image and analysing its randomness with human eyes. (Westfeld & Pfitzmann, 2000) clearly stated and proved that the least significant bits of luminance values of digital images are not completely random, whereas many authors wrongly assumed the complete randomness of the least significant bits in the image's luminance values. Therefore, the existence of the hidden message could be noticed by having a completely random noise, as shown in Figure 3.1 (Westfeld & Pfitzmann, 2000).

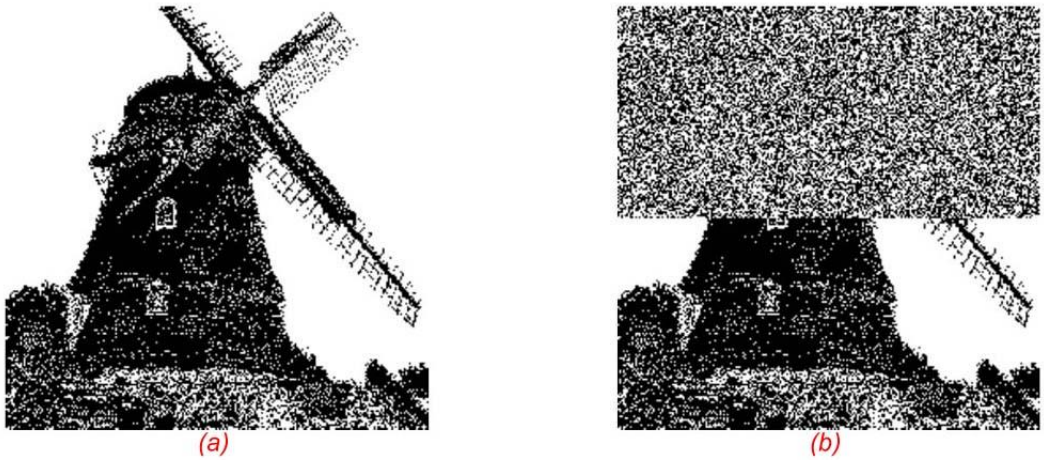


Figure 3.1: The visual attack; (a) is a clean image and (b) is a stego image with an embedding rate of 0.5

3.4.2 Structural Steganalysis

Embedding secret messages into any digital media leads to some changes in the data file format (or characteristic structure) (Wayner, 2002). Identifying these changes in the characteristic structure of any digital object can help the steganalyser to find the presence of hidden contents. However, structural attacks could be difficult because there are a range of physical structures to represent any digital media file, like texture, colour and contrast variance in normal images (Watters, Martin, & Stripf, 2005). Also, there are other circumstances that can affect the structural detection of hidden contents, like reducing the modification rate or using a larger cover file. Thus, in these cases statistical or visual attacks may perform better.

There are a number of structural steganalysis methods proposed to detect data embedding with LSB replacement in digital images, the most sensitive detection methods could be found in

previous studies (Dumitrescu, Wu, & Wang, 2003; Fridrich et al., 2001b; Fridrich, Goljan, & Soukal, 2003; Lu, Luo, Tang, & Shen, 2005; T. Zhang & Ping, 2003b). Based on having many commonalities among structural detectors of LSB embedding steganography in digital images, (A. Ker, 2005a) proposed a general framework of structural steganalysis for LSB replacement steganography that uses the structure or combinational properties of the LSB embedding method to detect and estimate the length of the hidden message.

3.4.3 Statistical Steganalysis

Mathematical statistics are used by scientists to determine whether some events occur at random or to prove their theory of explaining the event. As embedded messages are usually more random than the replaced information of the cover media (Wayner, 2002), many of these statistical methods can be used to indicate the availability of hidden content. Chi-square (X^2) is the simplest such statistical test to identify the randomness in an observed sequence of events. Chi-square attack (Westfeld & Pfitzmann, 2000) can detect the equally distributed least significant bits in digital image pixel values, which result from LSB embedding process and makes the frequency of each pair of values equal. Low scores indicate a high degree of randomness, which means there is a probability of having hidden contents.

Another statistical test of detecting LSB replacement in coloured images is measuring the number of close colours. These close colour pairs are different by maximum of one unit in each of their colour components (Red, Green, and Blue). Images with hidden content are expected to have more close pairs than clean ones (Fridrich & Long, 2000; N. Johnson & S. Jajodia, 1998; N. F. Johnson & S. Jajodia, 1998; Maes, 1998).

Basic LSB steganography could be detected with the previously stated statistical methods. However, more complicated LSB embedding methods can avoid this kind of analysis, for example if the embedding process does not affect the histogram of the image. (Sun, Chen, & Wang, 2006) used a certain method to swap PoVs with each other instead of flipping their least significant bits. The swapping process neither changes the statistical profile of the least significant bits nor the overall distribution of colours in the image (Wayner, 2002).

There are other more sophisticated statistical analyses that work by applying a number of functions that can be used to model images like wavelet functions (Buccigrossi & Simoncelli, 1999; Rinaldo & Calvagno, 1995; Shapiro, 1993), by which the coefficients of these wavelet decompositions could be analysed by finding the mean, variance, skewness and kurtosis to indicate the presence or absence of the hidden content (Wayner, 2002).

3.5 Steganalysis Types

As the detection of hidden contents can be modelled as a classification problem, it divides the given objects into two disjoined subsets of clean and stego objects. Some steganalysis methods are capable of detecting a group of steganography methods, and some steganalysis methods are designed to detect a specific embedding method. Thus, according to the range of detecting steganographic methods, the steganalysis can be classified into three main types; blind, semi-blind and targeted steganalysis (Cox et al., 2008).

3.5.1 Blind (or Universal) Steganalysis

If a warden has no prior information about the covert communication between Alice and Bob, except a certain level of suspicion, then she must develop such a steganalysis method that is capable of detecting all (or at least a wide range of) steganographic methods (Cox et al., 2008). Thus, the blind or universal steganalysis methods are intended to detect the existence of the secret message without having any prior knowledge about the embedding function (Lou, Chou, Tso, & Chiu, 2012). This type of steganalysis has the flexibility to be applied to different kinds of steganographic algorithms and also to be used in real-life situations like the analysis of digital forensics. However, they are less accurate in detection compared to targeted steganalysis schemes (Kharrazi, Sencar, & Memon, 2006). The fundamental concept of a blind steganalysis is to extract some features directly affected by message embedding, then classify the digital objects by using some classifiers.

The selection of statistical features is the key concern of designing any blind steganalysis algorithm. The most common typical statistical features are the probability of density function (PDF) moment and characteristic function (CF) moment (Xiangyang, Fenlin, Shiguo, Chunfang, & Gritzalis, 2011). Examples of such blind steganalysis methods include that proposed by Farid et al. (Farid, 2002), based on wavelet-like decomposition and PDF moments, and the accuracy of the method was improved by extracting features from three colour components of RGB images by (Lyu & Farid, 2003). Many studies have explored this area in depth (Goljan, Fridrich, & Holotyak, 2006; Gul & Kurugollu, 2010; Han, Fenlin, & Xiangyang, 2009; B. Li, Huang, & Shi, 2008; Y. Q. Shi et al., 2005; Xiaochuan, Yunhong, Tieniu, & Guo, 2006; Xuan et al., 2005).

3.5.2 Semi-Blind Steganalysis

If a warden is expecting the steganographic communication between Alice and Bob and has an idea about the possible steganographic algorithms they use, then she must have a special steganalaysis algorithm that can detect this range of steganographic schemes. The semi-blind (or semi-universal) steganalysis scheme could be applied on a selected set of steganographic

algorithms (Gireesh Kumar, Jithin, & Shankar, 2010). For example, steganalysis methods proposed in previous studies (Chunhua & Shi, 2008; Fridrich, 2005; Pevny & Fridrich, 2007; Y. Shi, Chen, & Chen, 2007) can accurately detect many JPEG steganographic schemes, but they may not be effective for spatial steganography (B. Li et al., 2011).

3.5.3 Targeted (or Specific) Steganalysis

If a warden is certain about the existence of covert communication between Alice and Bob and is also aware of the steganographic method they used, then she must develop a steganalysis method that is capable of detecting hidden messages embedded with their steganography method only. The targeted or specific steganalysis schemes use full knowledge of a specific (targeted) steganographic algorithm and are designed specifically to detect such a scheme (Cox et al., 2008). They are more reliable with better performance in detection than the universal schemes (Lou et al., 2012). Hence, many current targeted steganalysis methods are extended and could be classified as active steganalysis techniques, as they estimate the embedded message size. For example, these include LSB replacement steganalysis methods like regular and singular (RS) methods (Fridrich et al., 2001b), weighted stego (WS) (Fridrich & Goljan, 2004; Andrew D Ker & Böhme, 2008), sample pair analysis (SPA) (Dumitrescu et al., 2003), difference image histogram (DIH) (Tao & Xijian, 2003), and least squares method (LSM) (Lu et al., 2005).

3.6 Steganalysis Attacks

The possible attacks on steganographic schemes are similar to cryptography in terminology with some considerable technical differences. The steganalyst applies steganalysis methods in order to detect the existence of the secret message, whereas cryptanalyst applies the cryptanalysis methods to get the plain text from the encrypted version (Stefan Katzenbeisser & Petitcolas, 2000). However, if the secret message was encrypted before embedding, then the cryptanalysis methods may be applied, after retrieving the hidden information by a specific steganalysis method, if the retrieval of the message content is considered necessary. The steganalysis attack techniques are stego-only attack, known cover attack, known message attack, chosen stego attack, chosen message attack and known stego attack.

3.6.1 Stego-Only Attack

The stego-only attack could be considered the most realistic technique of practical steganalysis, by which the stego object is the only thing available to steganalysers (Stefan Katzenbeisser & Petitcolas, 2000). This type of attack relies on the content (or features) of the cover object and

utilises some specific statistical methods to find any artefact caused by the embedding process in relation to the expected normal cover type.

3.6.2 The Known Cover Attack

The known cover attack is a less common technique of steganalysis attack in which both cover and stego objects are available for steganalysers (Stefan Katzenbeisser & Petitcolas, 2000). The attacker should consider the difference between both digital media (cover and stego) to conclude some information about the embedding process and the location of the secret message. Nowadays, it is less applicable because there are millions of digital objects on the web, especially images, and also creating digital objects like images specifically for steganographic purposes became very easy with the increasing affordability of digital equipment.

3.6.3 Known Message Attack

The known message attack may be very difficult and considered like stego-only attack (Stefan Katzenbeisser & Petitcolas, 2000). It considers the availability of the hidden message, therefore the attacker should analyse the stego object to find the pattern that matches the hidden message for future attacks on the system.

3.6.4 Chosen Stego Attack

The chosen stego attack assumes that both the steganography algorithm and stego object are available to the attacker (Stefan Katzenbeisser & Petitcolas, 2000), thus the attacker may use them to find an estimated version of the hidden message.

3.6.5 Chosen Message Attack

Chosen message attack is considered as the most powerful attack (Lin & Delp, 1999), by which the steganalyst has access to the steganographic algorithm and can generate a stego object from his own message (Stefan Katzenbeisser & Petitcolas, 2000). The purpose of this type of attack is the determination of corresponding patterns in the stego object to be used for detecting a specific steganographic method.

3.6.6 Known Stego Attack

A known stego attack assumes that the attacker has access to both the original and stego objects and knows the embedding algorithm (Alturki & Mersereau, 2001). The scrambling key is not known by the attacker and the retrieval of the hidden message or discovering the key may be considered as the main goal for this type of attack.

3.7 Steganalysis as a Binary Classifier

Since the identification of embedded secret data in suspected media is the main goal of steganalysis, it can be modelled as a classification problem to determine whether the analysed media is stego or cover.

The general approach of classification can be defined as a task of dividing some objects into a number of disjoint classes in such a way that each object is classified to only one class, and no object remains unclassified (Max, 2007). More specifically, steganalysis could be considered as a binary classifier that classifies a set of digital media into two disjoint classes of stego and clean objects.

Before discussing the performance measurement of any steganalysis method, some details will be given about the classification model of steganalysis. The binary classifier of steganalysis considers two class labels: Positive (P) and Negative (N). These represent stego and clean classes, respectively. The steganalysis classifier maps each instance from Instances (I) to one element of the set $\{\mathrm{P}, \mathrm{N}\}$.

Based on the classification model, the output could be probabilistic (or continuous); the estimated probability of instance's class membership, or discrete class label of the instance's predicted class (Fawcett, 2003). Hence, according to the discrete class labels or specifying the detection threshold, there are four possible outcomes in this type of classification: true positives, false negatives, true negatives, and false positives, as explained in the following sections.

3.7.1 True Positives and False Negatives

Instances from actual positive (stego) class could be classified to either positive or negative classes by the classifier. True positive (TP) is the case when an actual positive instance is classified as positive by the classifier. False negative (FN) is the case when an instance is classified as negative by the classifier, while it is actually positive. All instances from true positives and false negatives belong to the actual positive (stego) class.

3.7.2 True Negatives and False Positives

Instances from actual negative (clean) class could be classified to either positive or negative classes by the classifier. True negative (TN) is the case when an actual negative instance is classified as negative by the classifier, and false positive (FP) is the case when an instance is classified as positive by the classifier, while it is actually negative. All instances from true negatives and false positives belong to the actual negative (clean) class.

3.7.3 Confusion Matrix

When a steganalysis method is applied on a set of testing instances of stego and cover media, a two-by-two confusion matrix (or contingency table) can be structured that represents the dispositions of the classified instances (Fawcett, 2003). The confusion matrix of steganalysis classification has two rows and columns, where the rows represent the actual classifications and the columns represent the predicted classifications (or detection results) (Max, 2007), as shown in Figure 3.2.

	Predicted Stego (P)	Predicted Clean (N)
Actual Stego (p)	True Positives (TP)	False Negatives (FN)
Actual Clean (n)	False Positives (FP)	True Negatives (TN)

Figure 3.2: Confusion matrix of a binary classifier (steganalysis)

So, the row-wise summation gives the total number of actual positive and negative instances of the testing data set, as in following;

$$p = TP + FN \qquad \textbf{(3.1)}$$

$$n = FP + TN \qquad \textbf{(3.2)}$$

Also, the column wise summation gives the total number of classified as positive and negative instances of the testing data, as in the following;

$$P = TP + FP \qquad \textbf{(3.3)}$$

$$N = FN + TN \qquad \textbf{(3.4)}$$

There are two types of classification errors; false positives (known as Type 1 error) and false negatives (known as Type 2 error). These two types of error are not always equally important and are highly dependent on the application. More details on this can be found in (Max, 2007).

The perfect classifier would have the value of (TP = p) and (TN = n). Other values would be (FP = 0) and (FN = 0). In other words, only leading diagonal entries of the confusion matrix would be non-zero.

3.8 Steganalysis Performance Evaluation

The confusion matrix forms the basis for many evaluation metrics. The most important measurements are the following (Max, 2007):

- True positive rate (or sensitivity) is the ratio of stego instances that are correctly classified as stego.

$$TPrate = \frac{TP}{p} \qquad \textbf{(3.5)}$$

- False positive rate (or false alarm rate) is the ratio of clean instances that are wrongly classified as stego.

$$FPrate = \frac{FP}{n} \qquad \textbf{(3.6)}$$

- True negative rate (or specificity) is the ratio of clean instances that are correctly classified as clean.

$$TNrate = \frac{TN}{n} \qquad \textbf{(3.7)}$$

- False negative rate is the ratio of stego instances that are wrongly classified as clean.

$$FNrate = \frac{FN}{p} \qquad \textbf{(3.8)}$$

- Precision (or positive predictive value) is the ratio of instances classified as stego that are really stego.

$$Precision = \frac{TP}{TP+FP} = \frac{TP}{P} \qquad \textbf{(3.9)}$$

- Accuracy (or predictive accuracy) is the ratio of instances that are correctly classified.

$$Accuracy = \frac{(TP+TN)}{(p+n)} \qquad \textbf{(3.10)}$$

- F1 Score is a combined measurement of precision and sensitivity.

$$F1\ Score = \frac{(2\times Precision\times Sensitivity)}{(Precision+Sensitivity)} \qquad \textbf{(3.11)}$$

- Error rate is the ratio of instances that are incorrectly classified.

$$Error\ rate = \frac{(FP+FN)}{(p+n)} \qquad \textbf{(3.12)}$$

The number of actual positive instances (p) and the number of actual negative instances (n) are fixed for a given set of testing instances. However, different classifiers will give different performance measurements when applied to the same set of instances. As can be observed from the above equations, all performance measurements could be found by knowing the true positive rate (TPrate) and false positive rate (FPrate), as well as the fixed values of positive (p) and negative (n) instances of the testing set. Therefore, the steganalysis classifier can be characterised by its true positive rate and false positive rate, in which their values range from 0 to 1 inclusively.

It is generally assumed that the predictive accuracy is the best (or only) way to measure the performance of the classifier. However, this is not necessarily the case, because it is derived from values in both rows of the confusion matrix, which are affected directly by the comparative size of

(p) and (n). In contrast, the TPrate and the FPrate values are independent from the comparative size of (p and n), as they are calculated from different rows of confusion matrix (Max, 2007).

For example, for the same classifier if we assume the confusion matrix is as in Figure 3.3, the true positive rate will be 0.89 and the false positive rate will be 0.2.

	Predicted Stego (P)	Predicted Clean (N)
Actual Stego (p = 9,000)	8,000	1,000
Actual Clean (n = 10,000)	2,000	8,000

Figure 3.3: The first example of confusion matrix

The predictive accuracy, according to equation 3.10, would be 0.842. However, increasing the number of positive instances from 9,000 to 90,000 would produce another confusion matrix like Figure 3.4.

	Predicted Stego (P)	Predicted Clean (N)
Actual Stego (p = 90,000)	80,000	10,000
Actual Clean (n = 10,000)	2,000	8,000

Figure 3.4: The second example of confusion matrix

So, the true positive rate and the false positive rate will not be affected by this, while the predictive accuracy (equation 3.10) will change from 0.842 to 0.88. Also, if we change the value of negative instances from 10,000 to 100,000 of the first example, another confusion matrix will be produced like in Figure 3.5.

	Predicted Stego (P)	Predicted Clean (N)
Actual Stego (p = 9,000)	8,000	1,000
Actual Clean (n = 100,000)	20,000	80,000

Figure 3.5: The third example of confusion matrix

The values of true positive rate and the false positive rate still remain unchanged, while the predictive accuracy will change from 0.842 to 0.807 for the same classifier.

Hence, the three predictive accuracies above will reflect changes in the comparative numbers of positive and negative values in the testing set of instances without changing the quality of the classifier in terms of TPrate and FPrate.

3.8.1 Receiver Operating Characteristic (ROC) Graph

The ROC ('Receiver Operating Characteristics') graph is a two-dimensional graph with a false positive rate plotted on the horizontal axis and the true positive rate plotted on the vertical access (Fawcett, 2003).

A discrete classifier only gives a class label for each instance of the testing set, and produces a single point in ROC space. Each point on the ROC graph (x,y) indicates the false positive rate as x and the true positive rate as y, whereby their values are between 0 and 1 inclusively. There are some important points on the ROC graph like: (0,0), (0,1), (1,0), and (1,1), each of which specifies a special case of the classifier (Max, 2007):

- The point (0, 1) represents the perfect classifier, which correctly classifies every instance of the testing set. Hence, its number of predicted true positive instances is equal to the actual number of positive instances (TP = p) and its number of predicted true negative instances is equal to the actual number of negative instances (TN = n).
- The point (1, 0) represents the worst possible classifier, which wrongly classifies every instance of the testing set. Hence, its number of predicted true positive instances is zero (TP = 0) and its number of predicted true negative instances is zero (TN = 0).
- The point (1, 1) represents the ultra-liberal classifier, which always predicts the positive class. Hence, its number of predicted true positive instances is equal to the actual number of actual positive instances (TP = p) and its number of predicted true negative instances is zero (TN = 0).
- The last point (0, 0) represents the ultra-conservative classifier, which always predicts the negative class. Hence, the number of predicted true positive instances is zero (TP = 0) and its number of predicted true negative instances is equal to the actual number of negative instances (TN = n).

Another important component of the ROC graph is the diagonal line that connects the points (0, 0) and (1, 1), which corresponds to the random guess. Any classifier performance point located on this line (points with equal values of false positive and true positive rates) indicates the random guess. Informally, any classifier located above the line of random guess is considered better than the random classifier. Also, the closer the classifier is to the perfect point (1, 0), the better it is. For example, in Figure 3.6 classifier A is better than classifier B, and they are both better than classifier C, which is no more than a random classifier.

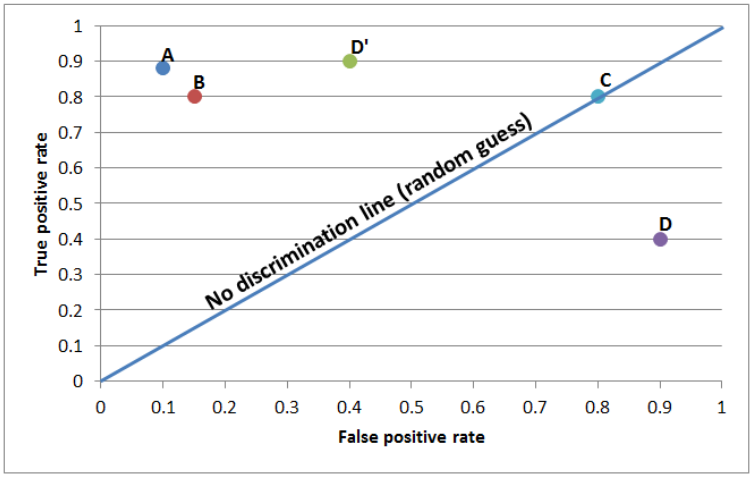


Figure 3.6: An example of ROC graph

Classifiers located under the line of random guess, like classifier D, which is worse than random guess, could be changed to a better classifier (classifier D' in Figure 3.6) by reversing its predictions in such a way that replaces the predicted positive instances with predicted negatives, and vice versa.

Some classifiers, like a Naive Bayes, normally produce a degree of membership for each instance as a numeric value known as probability or score of class membership. This type of probabilistic classifier could be used as a discrete classifier by setting a threshold value to output Yes or No labels for each instance of the testing set. Different threshold values, changing it from minimum possible value to the maximum possible value, produce different points in ROC space, which could be joined to form a ROC curve (Fawcett, 2003).

3.8.2 Finding the Best Classifier

For a given application, there is more than one method of finding the best classifier. The Euclidean distance is one of the approaches of measuring the performance of discrete classifiers; it measures the distance between the perfect classifier (0, 1) and the point that represents the performance of the given classifier on the ROC graph by the following equation (Max, 2007):

$$Euc = \sqrt{FPrate^2 + (1 - TPrate)^2} \qquad \textbf{(3.13)}$$

For the perfect classifier (0, 1), the Ecu is zero, and for the worst classifier (1, 0) it is $\sqrt{2}$. Hence, the smaller value of Euc represents a better classifier. However, it does not take the relative importance of true and false positives into account.

There is also a possibility to have a weighted version of the Euclidean distance equation like the following (Max, 2007):

$$WEuc = \sqrt{(1-w)FPrate^2 + w(1-TPrate)^2} \qquad \textbf{(3.14)}$$

Here, setting the value of w to zero reduces the weighted equation to $WEuc = FPrate$, which means minimizing the false positive rate is the only aim. Also, setting the value of w to 1 reduces the weighted equation to $WEuc = 1 - TPrate$, which means maximising the true positive rate is the only aim. Another possible value of w is 0.5, having equal weights for both.

For probabilistic classifiers that are usually represented by an ROC curve, the common method of comparing different classifiers is to calculate the area under the curve (AUC), whose value ranges between 0 and 1. The greater AUC represents a better average performance of the classifier. However, no realistic classifier should have an AUC value less than 0.5, as it is the AUC value of the line of random guess between the two points of (0, 0) and (1, 1) (Fawcett, 2003).

Figure 3.7 shows two classifiers A and B. It can be seen that the AUC of classifier B is greater than the AUC of classifier A. Note that the area of classifier B also exists under the area of classifier A. This shows that classifier B has a better performance than classifier A.

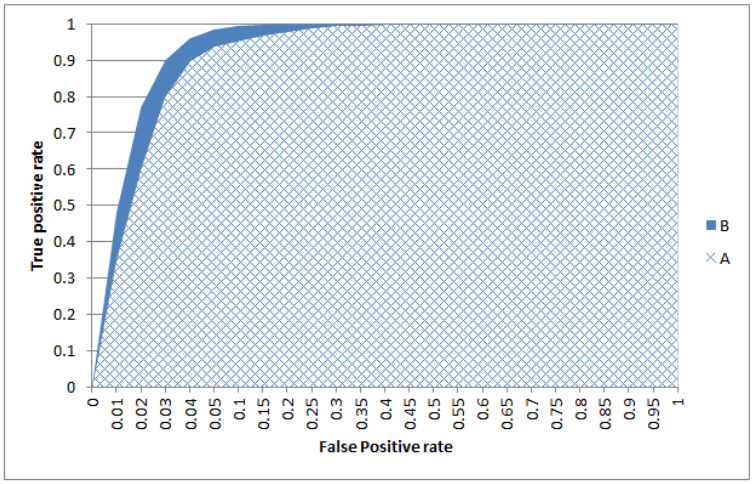


Figure 3.7: An example of ROC curve

3.9 Steganalysis and Digital Forensics

If the detection was not the only requirement and further tasks were needed, like extracting some attributes of the secret message or the embedding method, then forensic steganalysis is deployed. In some cases where other types of attack are not possible or not applicable, the forensic steganalysis would be the best. For example, after the warden was certain about the

steganographic communication between Alice and Bob, she may not have access to the resources to block the communication or does not want to break the communication in order to not let Alice and Bob know about the process of monitoring their communications (Cox et al., 2008).

The embedding algorithm could be known by examining the characteristics of embedding changes, and then it would be possible to determine the location of embedding and possibly extract at least the approximate version of the hidden message. In general, the amount of information known by the warden directly affects the possible type of attack, which in most cases would be stego-only attack. However, if more information were available to the warden, she could possibly perform further actions like known cover attack. For example, in the digital investigation process it is possible to have the suspect's computer for analysis, which may contain both cover and stego objects on the hard disk (Cox et al., 2008). Moreover, for law enforcement, especially cybercrime and copyright issues, the steganalysis tools are considered very important (Fridrich & Goljan, 2002).

3.10 Significant Steganalysis Algorithms of LSB Embedding

As the LSB embedding is the most common available steganographic method, it is important to describe three of the most well-known algorithms for both targeted and blind steganalysis in the spatial domain.

3.10.1 The Histogram Attack

The histogram attack is one of the early statistical steganalysis methods in the literature (Westfeld & Pfitzmann, 2000). It uses the characteristic artefacts left by LSB embedding in the histogram of pixel values. For instance, LSB embedding changes the even (2i) and the odd (2i+1) pixel values into each other; by adding 1 to even values and subtracting 1 from the odd values. This value transition happens when the LSB of the pixel value does not match the certain secret message bit. In general, it is expected that half of the pixel values will already contain the right value, and only the other half will change.

If h_s and h_c denote the number of pixels with the intensity value of n in the stego and clean images respectively, then for the relative message length of p, the expected stego intensity histogram could be determined by the following equation:

$$h_s(n) = \left(1 - \frac{p}{2}\right) h_c(n) + \frac{p}{2} \begin{cases} h_c(n+1)\,, n\ is\ even \\ h_c(n-1)\,, n\ is\ odd \end{cases} \qquad \textbf{(3.15)}$$

Of course, a random bit-stream is considered as a secret message, assuming that the secret message is encrypted or compressed (Cox et al., 2008). So, for the embedding rate of 1, the histogram bins will be very close for each pair of values (2i and 2i+1). Also, the sum of frequencies is invariant in each pair of values before and after embedding, as shown below:

$$h_s(2i) + h_s(2i+1) = h_c(2i) + h_c(2i+1) \qquad \textbf{(3.16)}$$

Without losing generality, the even values could be taken for a statistical test. Hence, the expected theoretical value for even valued histogram bins could be found by the following formula, for the embedding rate of 1:

$$\overline{h_s}(2i) = \big(h_s(2i) + h_s(2i+1)\big)/2 \qquad \textbf{(3.17)}$$

So, the Pearson's chi-square test is applied with $(k-1)$ degree of freedom, which is 127, to decide whether $h_s(2i) = \overline{h_s}(2i)$ or not, as shown below:

$$S = \sum_{i=1}^{k} \frac{[h_s(2i) - \overline{h_s}(2i)]^2}{\overline{h_s}(2i)} \qquad \textbf{(3.18)}$$

The unpopulated grey scales, $\overline{h_s}(2i) \leq 4$, will be omitted to make sure that S is approximately chi-square distributed. A small value of S implies the existence of LSB embedding and large values of S indicates that the image is clean and no message is embedded using the LSB method. The statistical significance of S is calculated by finding the p-value, as shown below:

$$P(S) = 1 - \frac{1}{2^{\frac{k-1}{2}}\,\Gamma\left(\frac{k-1}{2}\right)} \int_{s}^{\infty} e^{-\frac{x}{2}} x^{\frac{k-1}{2}-1} dx \qquad \textbf{(3.19)}$$

The p-value ranges between 0 to 1; 0 indicates that there is no hidden message, and 1 indicates that the image contains hidden message with the embedding rate of 1. This test can be used when the embedding is sequential and can also indicate the existence of the hidden message for bit rates of less than 1. This could be achieved by testing the image from 1% to 100% of the total pixels of the image. Hence, if 50% of the image contains hidden message, then the location and the embedding rate could be visualised as shown in Figure 3.8.

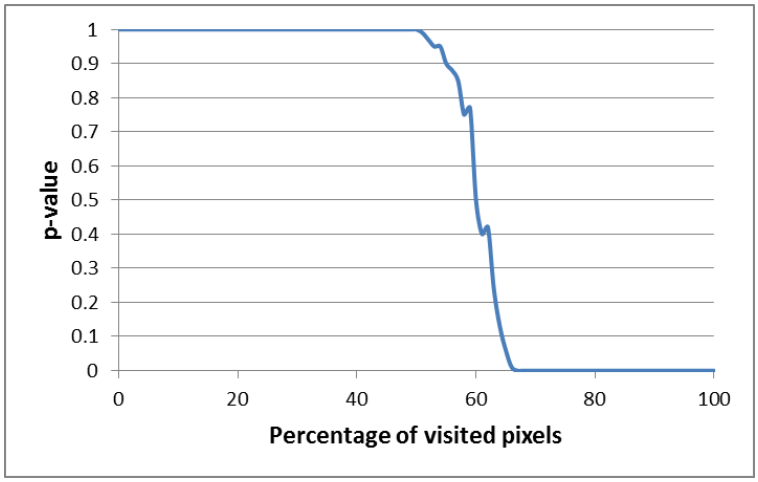


Figure 3.8: p-value vs. percentage of visited pixels for the embedding rate of 0.5

For pseudo-randomly selected pixels, this method stays efficient only when the majority of image pixels have been used by the embedding process. Therefore, this method has been generalised by calculating S using a sliding window over the pixel values of the image (N Provos & Honeyman, 2001).

Another method of extending the histogram attack to detect randomly distributed messages is examining the hashes (Westfeld, 2003) of small groups of adjacent pixels or colour components of a single pixel. This method can detect a random LSB embedding with an embedding rate of 0.3 and above.

3.10.2 Sample Pairs Analysis

The histogram attack on LSB steganography cannot be applied on stego images with low embedding rates of pseudo-randomly selected pixels, due to ignoring the neighbouring pixels' dependency in clean images. Hence, it is possible to utilise the spatial correlation in natural images for developing more reliable and accurate detection methods. A very good example is SPA (Dumitrescu, Wu, & Memon, 2002; Fridrich, Goljan, & Du, 2001a). SPA starts by declaring P as a set of all horizontally adjacent pixel pairs in the image, and then divides it into three disjoint subsets named X, Y, and Z where:

$$X = \{(u, v) \in P | \ (v \ is \ even \ and \ u < v) \ or \ (v \ is \ odd \ and \ u > v)\}$$

$$Y = \{(u, v) \in P | \ (v \ is \ even \ and \ u > v) \ or \ (v \ is \ odd \ and \ u < v)\}$$

$$Z = \{(u, v) \in P | \ u = v\}$$

Moreover, it divides Y into W and V, where:

$$W = \{(u, v) \in P |\, u = (2k, v = 2k + 1) or (u = 2k + 1, v = 2k)\}$$

$$V = Y - W$$

Now, the primary sets are $X, Y, W, V, \text{and } Z$, where:

$$P = X \cup W \cup V \cup Z$$

The LSB embedding will result in changing a given pixel pair (u, v) to transfer its membership to one of the primary sets according to the modification pattern, as shown in Table 3.1.

Table 3.1: Modification patterns

Modification pattern	Changes to (u, v)
00	Both values stay unmodified
01	Only v is modified
10	Only u is modified
11	Both values are modified

Since the relative number of modified pixels for LSB embedding with an embedding rate of q is $q/2$, then the probability of having a certain modification pattern is obtained by the following:

$$p(00, P) = \left(1 - \frac{q}{2}\right)^2$$

$$p(01, P) = p(10, P) = \frac{q}{2}\left(1 - \frac{q}{2}\right)$$

$$p(11, P) = \left(\frac{q}{2}\right)^2$$

Now, it is possible to denote the cardinalities of the primary sets after embedding of the relative message length of q.

$$|X'| = |X|\left(1 - \frac{q}{2}\right) + |V|\ \frac{q}{2}$$

$$|V'| = |V|\left(1 - \frac{q}{2}\right) + |X|\ \frac{q}{2}$$

$$|W'| = |W|\left(1 - q + \frac{q^2}{2}\right) + |Z|\ q\ \left(1 - \frac{q}{2}\right)$$

After assuming $|X| = |Y|$ for natural images, the embedding rate for a certain image could be found by obtaining the smaller root of the following quadratic equation:

$$\frac{y}{2}q^2 + (2|X'| - |P|)q + |Y'| - |X'| = 0$$

$$y = |W| + |Z| = |W'| + |Z'|$$

This detection method is further elaborated in previous studies (Böhme & Ker, 2006; Dumitrescu et al., 2003; A. Ker, 2005b; Andrew D. Ker, 2004; Andrew D Ker, 2007b; Lu et al., 2005), and it was extended to groups of more than two pixels by (Dumitrescu & Wu, 2005; A. Ker, 2005a). There are also some other related approaches in the literature (Chandramouli & Memon, 2001; Dabeer, Sullivan, Madhow, Chandrasekaran, & Manjunath, 2004; Fridrich & Goljan, 2004; Fridrich, Goljan, & Soukal, 2003; K. Lee, Westfeld, & Lee, 2006; Xiaopi, Yunhong, & Tieniu, 2004; T. Zhang & Ping, 2003a, 2003b).

3.10.3 Blind Steganalysis in the Spatial Domain

The spatial domain and JPEG domain blind steganalysis are similar, but they use different features; for example, the calibration based method of JPEG domain cannot be applied for spatial domain, rather noise reduction filters are used (Kivanc Mihcak, Kozintsev, Ramchandran, & Moulin, 1999).

The spatial domain steganographic methods can be illustrated as adding noise with certain properties (Cox et al., 2008). Hence, the histogram of the stego image is a convolution of both probabilities of mass function of the cover and noise signals. As the noise signals represent a low pass filtering on the histogram of the image, the histogram of the stego signal will be smoother than the histogram of the cover signal, and the low frequencies will have more concentrated energy.

Thus, the histogram characteristic function (HCF) of the stego image is represented by the Fourier-transformed values of the histogram.

$$H_s[k] = H_c[k]\, F[k] \quad \text{for each } k$$

Where H_s and H_c represent the HCFs of stego and cover images respectively, and F is the Fourier representation of the probability mass function of the added noise signal.

There are a number of steganalysis methods based on HCF (Harmsen & Pearlman, 2003; A. D. Ker, 2005a, 2005b), and other features used by blind steganalysis methods have been demonstrated (Avcibaş, Kharrazi, Memon, & Sankur, 2005; Lyu & Farid, 2003; Westfeld, 2003; Xuan et al., 2005).

3.11 Summary

The main purpose of steganalysis is the detection of hidden messages in digital media files. Further requirements like recovering the attributes of the hidden message (its length or content) and identifying the stego key and embedding method are the aim of forensic steganalysis. Generally, there are two approaches for detecting LSB steganography. One is to use the specific structural properties like LSB embedding method, for example sample pairs (Dumitrescu et al., 2003), the pairs analysis (Fridrich, Goljan, & Soukal, 2003), and difference histogram (T. Zhang & Ping, 2003b). These methods consider pairs of pixels with different selection schemes (A. Ker, 2005a). Other detectors rely on extracting the feature vectors using signal processing techniques for some sort of learning machine. This approach could be a very simple noise detector (Harmsen & Pearlman, 2003) or a sophisticated wavelet method (Lyu & Farid, 2004).

There are three main classes of steganalysis methods: blind (or universal), semi-blind and targeted. The universal steganalysis methods can detect a wide range of embedding methods. However, they are less accurate than targeted methods and do not give any information about the characteristics of the hidden message (A. Ker, 2005a).

Both histogram attack and sample pairs analysis can estimate the length of the hidden message. However, the histogram attack can accurately detect the sequential LSB embedding, whereas sample pairs analysis can detect pseudo-randomly distributed LSB embedding. Moreover, passive steganalysis is the most common type steganalysis technique, whereas the most realistic steganographic attack technique is the stego-only attack.

All steganalysis methods can be modelled as a classification problem. Hence, tools like pattern recognition and machine learning can also be applied. The steganalysis classifier can be characterised by its true positive and false positive rates, as they are not affected by the comparative size of positive and negative instances in the testing set. Both true positive and false positive rates can be visualised in a two dimensional ROC graph, to determine the performance of the classifier.

CHAPTER 4: SINGLE MISMATCH STEGANOGRAPHY

4.1 Introduction

LSB replacement is the most widely used embedding method because it is extremely easy to implement, has a reasonable capacity, and is visually imperceptible. However, it can reliably be detected by current steganalysis methods, so modified versions of the LSB replacement method have been proposed by steganographers to reduce the probability of detection and improve their capacity. The extensions of LSB replacement that have received great attention from steganographers focus on two least significant bits (2LSB) replacement, because this replacement method is still visually imperceptible, has a higher capacity than LSB embedding, is very easy to implement and results in more complex changes to the intensity histogram of the cover image. However, 2LSB replacement is detectable by some steganalysis methods. Additionally, high bit-planes in the embedding process will degrade the quality of the stego image and negatively affect its visual imperceptibility.

Since the probability of detection is highly dependent on the amount of changes in the cover image as a result of inserting a larger amount of noise, developing an embedding method with fewer changes to the cover image pixel values for the same amount of secret data is considered important to reduce the probability of detection in both LSB and 2LSB steganography of digital images. In this chapter, a new method of both LSB and 2LSB steganography (which depends on the match/mismatch cases) is proposed in still images to improve the embedding efficiency and reduce the probability of detection by their targeted steganalysis methods. Moreover, the proposed method results in less bit-level changes on the pixel values of the cover image and modifies the intensity histogram in a different way.

The single mismatch, we proposed in this chapter, always creates single mismatch between two bits of the secret message and the LSBs of the image pixel values. We are going to demonstrate that the proposed method outperforms the security of different methods of LSB and 2LSB steganography by reducing the probability of detection with their current targeted steganalysis methods. Other advantages of the proposed method include reducing the overall bit-level changes to the cover image for the same amount of embedded data and avoiding complex calculations. However, the new method results in small additional distortion in the stego image, which could be tolerated as discussed in later sections.

This chapter is organised as follows; it starts by introducing LSB steganography and steganalysis, and clarifying adaptive and non-adaptive LSB steganography in images, then it explains the concept of improving the embedding efficiency and its effect on the probability of detection and analyses different methods of non-adaptive LSB steganography in digital images. Later, a new proposed method of LSB steganography is discussed with its analysis, experimental results, and extraction process. After that, the 2LSB steganography and steganalysis are explained. Then, the concept of improving the embedding efficiency of 2LSB steganography is explained and the 2LSB replacement is analysed in detail. This is followed by the new proposed method with its analysis, experimental results and extraction process. Then, the conclusions of both proposed methods (for LSB and 2LSB steganography) are presented.

4.2 LSB Steganography

As mentioned earlier, LSB steganography is the most widely used embedding method in pixel domain, since it is easy to implement, has reasonable capacity, and is visually imperceptible. LSB replacement takes the selected pixel value and replaces its LSB value with 1-bit of the secret message. Since there is a probability of 50% that the LSB of the selected pixel value contains the desired bit value, it leaves half of the pixel values unmodified during the embedding process. Another well-known LSB steganography method is the LSB matching (±1 embedding). It is a modified version of the LSB replacement, where instead of simply replacing the LSB of the selected pixel value with the value of 1-bit of the secret message, the LSB matching randomly increases or decreases the pixel value if its LSB value does not match the value of the secret message bit. This again leaves the pixel value unmodified if its LSB value matches the value of the secret message bit.

Unfortunately, both methods of LSB steganography (replacement and matching) are detectable by the current steganalysis approaches discussed in later sections. Therefore, some methods have been proposed to improve the capacity of LSB replacement based on pixel value differences and LSB replacement (D.-C. Wu & Tsai, 2003; H.-C. Wu, Wu, Tsai, & Hwang, 2005). To avoid changing the histogram of the cover image, another method was proposed by (Sun et al., 2006) using rearrangement and swapping phases, which reduce the embedding capacity by 50%. However, this method negatively affects the hiding capacity and the level of distortion, which affects the undetectability of this method.

4.3 LSB Steganalysis

The imbalance distortion of LSB replacement data embedding causes the 'Pairs of Values' to appear in the intensity histogram of the stego image. This property could serve steganalysers a lot and enable them to propose accurate detection methods. Moreover, the LSB replacement is inherently asymmetric; there are many steganalysis methods that can reliably detect them (Andrew D Ker, 2007a), like regular-singular (RS) (Fridrich et al., 2001a), sample pairs (SP) (Dumitrescu et al., 2003) discussed in section 3.10.2, and weighted stego image (WS) (Fridrich & Goljan, 2004; Andrew D Ker & Böhme, 2008) analyses.

The RS method proposed by (Fridrich et al., 2001a), divides an image into disjoint groups of 4-neighboring pixels. They use the discrimination function (variation) to capture the smoothness in the pixel groups. They use two invertible operations F called flipping; F_1 as $0\leftrightarrow1$, $2\leftrightarrow3$, …, $254\leftrightarrow255$ and F_{-1}: $-1\leftrightarrow0$, $1\leftrightarrow2$, …, $255\leftrightarrow256$. In addition, they use an identity operation F_0. Based on the discrimination function of the different flipped pixel groups, the pixel groups are divided into Regular, Singular and Unused groups.

RS is a reliable method of detecting non-sequentially LSB embedding based on regular and singular groups. According to experimental results (Fridrich et al., 2001a), the upper bound of 0.005 bits/ pixel is considered safe in terms of detection. The SP analysis (Dumitrescu et al., 2003) can accurately detect the LSB replacement embedding method based on a finite state machine of trace multi-sets, since flipping the LSBs will transfer these multi-sets into each other with given probabilities. Therefore, the statistical relations between the cardinalities of trace multi-sets will change. The concept of weighted stego image (Fridrich & Goljan, 2004; Andrew D Ker & Böhme, 2008) can detect and estimate the size of the hidden message embedded with LSB replacement in random pixel positions. The problem of estimating the message length is formulated as a simple optimisation problem. The WS image steganalysis method was improved by (Andrew D Ker & Böhme, 2008) to detect a sequentially located payload by upgrading the three components of the detection method; the prediction of cover pixel, least-squares weight, and bias correction.

Apart from the supervised machine learning detectors of LSB matching (or ±1 embedding) used by numerous studies (Giacomo Cancelli et al., 2008; Fridrich, Soukal, et al., 2005; Goljan et al., 2006; Sullivan, Madhow, Chandrasekaran, & Manjunath, 2006), which usually have problems in choosing an appropriate feature set and measuring classification error probabilities (Cogranne & Retraint, 2013), the methods of detecting LSB matching steganography could be divided into two categories: the centre of mass of the HCF and the amplitude of local extrema (ALE) (G. Cancelli, Doerr, Barni, & Cox, 2008).

A number of detection methods have been proposed based on the centre of mass of the histogram characteristic function (HCF-COM). That of (Harmsen & Pearlman, 2003) has better performance for RGB images than grey-scale. They relied on the fact that the LSB embedding method is equivalent to a low-pass filtering of the image histogram. This method was then modified and improved by Ker (A. D. Ker, 2005b), who applied the HCF in two novel ways: using the down sampled images and computing the adjacency histogram.

Based on the ALE (Amplitude of Local Extrema), a targeted steganalysis method is proposed (J. Zhang et al., 2007) based on the fact that after applying LSB matching the local maxima of the image histogram will decrease and the local minima will increase. So, it considers the summation of the absolute differences of local extrema and their neighbours in the intensity histogram, which its value is expected to be smaller for stego images than the clean one. This method was improved by (Giacomo Cancelli et al., 2008) after reducing the border effects of noise in the histogram and extending it to the ALE in the 2D adjacency histogram.

4.4 Adaptive and Non-Adaptive LSB Steganography in Images

The embedding process of LSB steganography relies on some methods for selecting the location of the change. In general, there are three selection rules to follow in order to control the location of change, which are either sequential, random or adaptive (Cox et al., 2008).

A sequential selection rule modifies the cover object elements individually by embedding the secret message bits in a sequential way. For example, it is possible to embed the secret message by starting from the top-left corner of an image to the bottom-right corner in a row-wise manner. This selection rule, known as sequential, is very easy to implement, but has very low security against detection methods.

A pseudo-random selection rule modifies the cover object by embedding the secret message bits into a pseudo-randomly chosen subset of the cover object, possibly by using a secret key as a PRNG. This type of selection rule gives a higher level of security than sequential methods.

An adaptive selection rule modifies the cover object by embedding the secret message bits in selected locations based on the characteristics of the cover object. For example, choosing noisy and highly textured areas of the image would be less detectable than smooth areas for hiding data. This selection rule, known as adaptive, gives higher security than sequential and pseudo-random selection rules in terms of detection.

Hence, the non-adaptive image steganography techniques modify the cover image for message embedding without considering its features (content). For example, LSB replacement and LSB matching with sequential or random selection of pixels modify the cover image according to the secret message and the key of random selection of pixels without taking the cover image properties into account. On the other hand, adaptive image steganography techniques modify the cover image in correlation with its features (Fridrich & Du, 2000). In other words, the selection of pixel positions for embedding is adaptive, depending on the content of the cover image. The bit-plane complexity segmentation (BPCS) proposed by (Kawaguchi & Eason, 1999) is an early typical method of adaptive steganography.

As adaptive steganographic schemes embed data in specific regions (such as edges), the steganographic capacity of such methods is highly dependent on the cover image used for embedding. Therefore, in general the adaptive schemes are expected to have less embedding rate than non-adaptive schemes. However, steganographers have to pay this price in order to have better security or less detectable stego images.

4.5 Improving the Embedding Efficiency and Undetectability of LSB

Undetectability is the most important requirement of any steganographic scheme, which is affected by the choice of the cover object, the type of embedding method, the selection rule of modifying places, and the number of embedding changes which are directly related to the length of secret message (Fridrich & Soukal, 2006).

If two different embedding methods share the same source of cover objects, the same selection method of embedding place, and similar embedding operation, the one with less number of embedding changes will be more secure (i.e. less detectable) because the statistical property of the cover object is less likely to be disrupted by a smaller number of embedding changes (Fridrich & Soukal, 2006).

The matrix encoding proposed by (Crandall, 1998), is one of the first attempts to reduce the number of changes during the embedding process. However, it limits the embedding capacity to 67% and is not useful for the embedding rate of 1.

The concept of embedding efficiency was introduced by (Westfeld & Pfitzmann, 2001), then considered as an important feature of steganographic schemes (Fridrich, Goljan, & Soukal, 2005; SALLEE, 2005), which is the expected number of embedded random message bits per single embedding change (Fridrich, Lisoněk, et al., 2007).

Reducing the expected number of modifications per pixel (ENMPP) is well studied in the literature, considering the embedding rate of less than 1, like Westfeld's F5-algorithm (Westfeld, 2001), which implements the matrix encoding to improve the efficiency of embedding. It uses permutative straddling to uniformly distribute the changes over the stego image. This method could improve the embedding efficiency only for short messages. However, short messages are already challenging to detect. Also, the source coding-based steganography (matrix embedding) proposed by Fridrich et al. (Fridrich, Lisoněk, et al., 2007; Fridrich & Soukal, 2006), which is an extension of F5-algorithm, improves the embedding efficiency for large payloads, but still with an embedding rate of less than 1. The stochastic modulation proposed by (Fridrich & Goljan, 2003) is another method of improving the security for the embedding rate of up to 0.8 bits/ pixel.

Another method proposed by (Chan, 2009), to improve the embedding efficiency using a binary function for the consecutive pixels. However, this improvement relies on the cover image and secret message properties, which does not perform equally for different embedding cases.

For the embedding rate of 1, there have been some methods for improving the embedding efficiency of LSB matching, like that of (Mielikainen, 2006), which reduced the ENMPP with the same message length from 0.5 to 0.375. The choice of whether to add or subtract one to/from a pixel value of their method relies on both the original pixel values and a pair of two consecutive secret bits. However, this method of embedding cannot be applied on saturated pixels (i.e. pixels with values 0 to 255), which is one of the drawbacks of the method. The generalisation method of LSB matching is proposed by (X. Li et al., 2009) with the same ENMPP for the same embedding rate using sum and difference covering set (SDCS), which is again has limitation when the pixel value is 0 or 255. Another method of improving the embedding efficiency of LSB matching is proposed by (W. Zhang, Zhang, & Wang, 2007) using a combination of binary codes and wet paper codes. The embedding efficiency of this method can achieve the upper bound of the generalised ±1 embedding schemes.

(Iranpour & Farokhian, 2013), also proposed an embedding method to improve the embedding efficiency using three binary functions to embed three bits of the secret message in three pixel values of the cover image. The ENMPP is 0.375 for the proposed method, but it has limitations in saturated pixel values (0 and 255). Hence, the maximum embedding capacity would be less than the LSB replacement.

So, no method could be found in literature to improve the embedding efficiency of non-adaptive LSB embedding for the *embedding rate of 1*. Here we focus on developing an embedding method

that can achieve the ENMPP of 0.375 for the embedding rate of 1. Developing such a method could be more useful than other adaptive methods from a usability perspective. Moreover, the non-adaptive LSB embedding methods with higher embedding efficiency can be used by existing adapted embedding methods to improve the steganographic capacity and reduce the probability of detection. A good example of non-adaptive methods is the LSB matching revisited (Mielikainen, 2006), which was extended in other studies (Huang, Zhao, & Ni, 2011; Kumar & Shunmuganathan, 2012; Weiqi, Fangjun, & Jiwu, 2010).

In this chapter, a new method of non-adaptive LSB steganography is proposed to reduce the probability of detection for the same amount of data embedded with LSB replacement, LSB matching, and LSB matching revisited (Mielikainen, 2006) by the current detection methods. The proposed method also results in fewer ENMPP in both pixel and bit-level to the cover image, and changes the histogram of the cover image in a different way, without any complex calculations.

4.5.1 Analysis of LSB Replacement

In this section, LSB replacement is analysed in three perspectives: the embedding process itself (with its embedding efficiency), its effect on the intensity histogram after embedding process, and the bit-level ENMPP for each bit of the secret message. Also, the main weaknesses of this embedding method are highlighted with the steganalysis methods that can detect it.

LSB replacement steganography simply replaces the LSB of the cover image pixel value with the value of a single bit of the secret message. It leaves the pixel values unchanged when their LSB value matches the bit value of the secret message and changes the mismatched LSB by either incrementing or decrementing the even or odd pixel values by one respectively (Mielikainen, 2006), as shown in Figure 4.1 for grey-scale image pixel values.

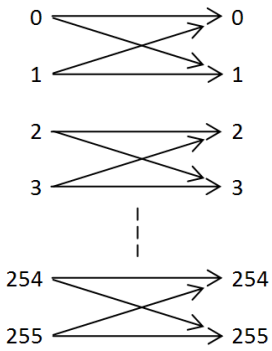


Figure 4.1: Possible pixel value transitions with LSB replacement

Therefore, the embedding algorithm of the LSB replacement can be formally described as follows:

$$P_s = \begin{cases} P_c + 1 & , \text{if } b \neq LSB(P_c) \text{ and } P_c \text{ is even} \\ P_c - 1 & , \text{if } b \neq LSB(P_c) \text{ and } P_c \text{ is odd} \\ P_c & , \text{if } b = LSB(P_c) \end{cases} \quad \textbf{(4.1)}$$

Where, P_s and P_c represent the stego and cover image pixel values respectively, and b is the desired bit value of the secret message.

To analyse the influence of the LSB replacement on the cover image intensity histogram, we should consider that there is a probability of 50% for the LSB of the cover image pixel value that already have the desired value. Therefore, the probability of modified pixel values will be (p/2) for an embedding rate of p and the unmodified pixel values will be (1-p/2) after the embedding process, which means that embedding each message bit needs 0.5 pixel values to be changed. In other words, it has an embedding efficiency of 2-bits of the secret message per one embedding change. Hence, the intensity histogram, using equation 4.1, of the stego image could be estimated as follows:

$$h_s(n) = \left(1 - \frac{p}{2}\right) h_c(n) + \frac{p}{2} \begin{cases} h_c(n+1) & , n \text{ is even} \\ h_c(n-1) & , n \text{ is odd} \end{cases} \quad \textbf{(4.2)}$$

Where n is a grey-scale level which ranges from 0 to 255, and $h_s(n)$ and $h_c(n)$ indicate the number of pixels in the stego and cover images respectively, with grey-scale value of n.

This type of embedding leads to an imbalance distortion and produces 'Pairs of Values' on the intensity histogram of the stego image. Since LSB replacement is inherently asymmetric, current steganalysis methods can detect it easily (Andrew D Ker, 2007a), including RS (Fridrich et al., 2001a), SP (Dumitrescu et al., 2003), and WS (Fridrich & Goljan, 2004; Andrew D Ker & Böhme, 2008).

Another way of analysing LSB embedding is the bit-level ENMPP, which is the expected number of bit modifications per pixel. This is also important, as there are some steganalysis methods that can detect the existence of the secret message based on calculating several binary similarity measures between low bit-planes (Avcibaş et al., 2005). Hence, an embedding process with less bit-level ENMPP would be less detectable by such detection methods.

The overall bit-level ENMPP for LSB replacement could be estimated by multiplying the probability of having mismatched LSBs, $P_r(\overline{M})$, which is 0.5 by the number of bits that needs to be changed in each case:

$$\text{bit} - \text{level ENMPP} = P_r(\overline{M}) \times \text{no. of modified bits} \qquad \textbf{(4.3)}$$

$$\text{bit} - \text{level ENMPP} = 0.5 \times 1 = 0.5 \quad \text{bits per message bit}$$

Hence, the overall bit-level ENMPP for LSB replacement is 0.5 bits for each bit of the secret message.

4.5.2 Analysis of LSB Matching (± Embedding)

To analyse LSB matching steganography, the embedding process is considered (with its embedding efficiency), its effect on the intensity histogram of the cover image, and its bit-level ENMPP.

LSB matching, or ±1 embedding, is a modified version of LSB replacement. Instead of simply replacing the LSB of the cover image, it randomly either adds or subtracts the cover image pixel value by one for LSB mismatched with the secret message bit (Sharp, 2001). The possible pixel value transitions of ±1 embedding are shown in Figure 4.2.

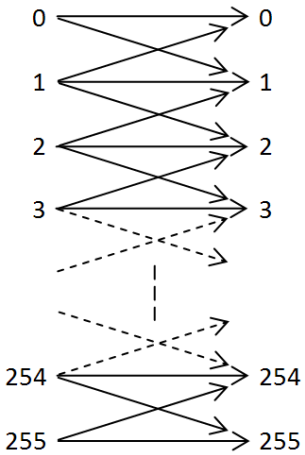


Figure 4.2: Possible pixel value transitions with LSB matching

The random increment or decrement in pixel values should maintain the boundary limitation, and pixel values should always remain between 0 and 255 (A. D. Ker, 2005b). In other words, the embedding process should neither subtract 1 from pixel values of 0 nor add 1 to the pixel values of 255.

This random ±1 change to the mismatched LSB pixel values avoids the asymmetry changes to the cover image, which is the case with LSB replacement. Hence, LSB matching is considered harder to

detect than LSB replacement (Mielikainen, 2006). The embedding procedure of LSB matching can be formally represented as follows (Xi, Ping, & Zhang, 2010):

$$P_s = \begin{cases} P_c + 1 & , \text{if } b \neq LSB(P_c) \text{ and } (K > 0 \text{ or } P_c = 0) \\ P_c - 1 & , \text{if } b \neq LSB(P_c) \text{ and } (K < 0 \text{ or } P_c = 255) \\ P_c & , \text{if } b = LSB(P_c) \end{cases} \quad \textbf{(4.4)}$$

Where P_s and P_c represent the stego and cover image pixel values respectively, and K is an independent and identically distributed random variable with uniform distribution on $\{-1, +1\}$.

For the intensity histogram we consider an embedding rate of p. There is a chance of 50% that the clean image pixel value contains the desired LSB, which means that $(p/2)$ of the cover pixel values will change after the embedding process. Hence, the estimated unmodified pixel values will be $(1 - p/2)$, which means that embedding each message bit needs 0.5 pixel values to be changed. In other words, its embedding efficiency is 2-bits of the secret message per one embedding change. The intensity histogram of the stego image could be obtained as follows (Xi et al., 2010), using equation 4.4:

$$h_s(n) = \left(1 - \frac{p}{2}\right) h_c(n) + \frac{p}{4}[h_c(n+1) + h_c(n-1)] \quad \textbf{(4.5)}$$

As mentioned earlier, the LSB matching will avoid the asymmetric property in modifying the cover image. However, as claimed by (J. Zhang et al., 2007), ±1 embedding is reduced to a low pass filtering of the intensity histogram. This implies that the cover histogram contains more high-frequency power than the histogram of the stego image (Xi et al., 2010), which offers an opportunity to steganalysers to detect the existence of the secret message embedded with LSB matching.

The methods of detecting ±1 embedding, excluding the supervised machine learning detectors, are divided into two main categories; the centre of mass of the HCF and the ALE (G. Cancelli et al., 2008).

The bit-level ENMPP of LSB matching is also important and should be considered, especially for steganalysis methods like binary similarity measures. Since the probability of having mismatched LSB is also 50%, based on the equation 4.3, the bit-level ENMPP would be as follows:

$$\text{bit} - \text{level ENMPP} = 0.5 \times (\geq 1)$$

$$\text{bit} - \text{level ENMPP} \geq 0.5 \text{ (bits per message bits)}$$

Where $P_r(\overline{M})$ is the probability of having mismatched LSBs, which is 0.5. However, the number of modified bits would be more than 1, because of the random ±1 changes to the pixel values, as noted from the following examples:

127 $(0111111)_2$ + 1 = 128 $(10000000)_2$, 8-bits changed

192 $(11000000)_2$ - 1 = 191 $(10111111)_2$, 7-bits changed

7 $(00000111)_2$ + 1 = 8 $(00001000)_2$, 4-bits changed

240 $(11110000)_2$ - 1 = 239 $(11101111)_2$, 5-bits changed

Hence, the overall bit-level ENMPP for LSB matching is expected to be more than or equal to 0.5 bits for each bit of the secret message.

4.6 Single Mismatch LSB Steganography (SMLSB)

Based on highlighting the weakness of both LSB replacement and ±1 embedding, in this section a new method of LSB embedding is proposed to improve the embedding efficiency and reduce the probability of detection by current targeted steganalysis methods. Moreover, the new proposed method should also minimize the bit-level ENMPP to the cover image after embedding.

The proposed method, single mismatch LSB embedding (SMLSB), takes two bits of the secret message at a time and embeds them in a pair of selected pixel values of the cover image. The embedding method creates a single mismatch between the 2-bits of the secret message and the LSBs of the selected pair of pixel values. For each 2-bits of the secret message, two consecutive pixel values are considered for simplicity. However, the selection could be based on other functions as well.

Since the proposed method embeds 2-bits at a time, there are four cases of having match (M) or mismatch ($\overline{M}$) between the LSBs of the selected two pixel values and the 2-bits of the secret message, as shown in Figure 4.3.

	LSB
Pixel value 1	M
Pixel value 2	M

	LSB
Pixel value 1	M
Pixel value 2	$\overline{M}$

	LSB
Pixel value 1	$\overline{M}$
Pixel value 2	M

	LSB
Pixel value 1	$\overline{M}$
Pixel value 2	$\overline{M}$

Figure 4.3: The possible cases of Match/ Mismatch

As the embedding method creates a single mismatch ($\mathrm{M}\overline{\mathrm{M}}$ or $\overline{\mathrm{M}}\mathrm{M}$) between pixel values and secret message bits, the 2nd LSB of the first pixel value should refer to the index of the mismatch; 1 for $\mathrm{M}\overline{\mathrm{M}}$ and 0 for $\overline{\mathrm{M}}\mathrm{M}$. If both LSB values are matched with the 2-bits of the secret message, the case is MM; then it changes one of the LSBs according to the value of 2nd LSB of the first pixel value. If the 2nd LSB value was 0, then it flips the LSB of the first pixel value to create $\overline{\mathrm{M}}\mathrm{M}$. Otherwise, if it was 1, it flips the LSB of the second pixel value to create $\mathrm{M}\overline{\mathrm{M}}$. For the $\overline{\mathrm{MM}}$ case, where both LSB values are mismatched with the 2-bits of the secret message, the embedding will also change one of the LSBs according to 2nd LSB of the first pixel value, but this time, if the 2nd LSB value was 0, then it flips the LSB of the second pixel value to create $\overline{\mathrm{M}}\mathrm{M}$. Otherwise, if it was 1, it flips the LSB of the first pixel value to create $\mathrm{M}\overline{\mathrm{M}}$.

For the other two cases, $\mathrm{M}\overline{\mathrm{M}}$ and $\overline{\mathrm{M}}\mathrm{M}$, the embedding will be done by changing the 2nd LSB of the first pixel value based on the index of the mismatch. If it was $\mathrm{M}\overline{\mathrm{M}}$, then the 2nd LSB of the first pixel value will be set to 1. Otherwise, if it was $\overline{\mathrm{M}}\mathrm{M}$, then the 2nd LSB value of the first pixel value will be set to 0. Hence, after each embedding there is only $\mathrm{M}\overline{\mathrm{M}}$ or $\overline{\mathrm{M}}\mathrm{M}$ with the right index in the 2nd LSB of the first pixel value. The embedding algorithm is shown in Figure 4.4. Table 4.1 shows some examples of the embedding process by the proposed method.

input: two cover pixel values x_1, x_2, and two message bits b_1, b_2

output: stego pixel values y_1, y_2

$y_1 := x_1$

$y_2 := x_2$

if $LSB(x_1) = b_1$ *AND* $LSB(x_2) = b_2$

{

if $2^{nd} LSB(x_1) = 0$

$LSB(y_1) := \overline{b_1}$

else

$LSB(y_2) := \overline{b_2}$

}

else if $LSB(x_1) \neq b_1$ *AND* $LSB(x_2) \neq b_2$

{

if $2^{nd} LSB(x_1) = 0$

$LSB(y_2) := \overline{b_2}$

else

$LSB(y_1) := \overline{b_1}$

}

else if $LSB(x_1) = b_1$ *AND* $LSB(x_2) \neq b_2$

$2^{nd} LSB(y_1) := 1$

else if $LSB(x_1) \neq b_1$ *AND* $LSB(x_2) = b_2$

$2^{nd} LSB(y_1) := 0$

end

Figure 4.4: The embedding algorithm of SMLSB embedding

Table 4.1: Examples of SMLSB embedding process

Clean pair of pixels	Two message bits	Stego pair of pixels
xxxxxx01 xxxxxxx1	11	xxxxxx0**0** xxxxxxx1
xxxxxx11 xxxxxxx0	10	xxxxxx11 xxxxxxx**1**
xxxxxx01 xxxxxxx1	00	xxxxxx01 xxxxxxx**0**
xxxxxx11 xxxxxxx0	01	xxxxxx1**0** xxxxxxx0
xxxxxx11 xxxxxxx0	11	xxxxxx11 xxxxxxx0
xxxxxx01 xxxxxxx1	10	xxxxxx**1**1 xxxxxxx1
Xxxxxx11 xxxxxxx1	01	xxxxxx**0**1 xxxxxxx1
xxxxxx00 xxxxxxx0	10	xxxxxx00 xxxxxxx0

4.6.1 Analysis of SMLSB Embedding

To analyse the proposed LSB embedding, just like other embedding methods mentioned earlier, the embedding process itself (with its embedding efficiency) is considered, its effect on the intensity histogram of the image, and the bit-level ENMPP as well.

SMLSB embedding modifies the pixel values based on the match/mismatch cases between LSBs of the selected two pixel values and the 2-bits of the secret message. As it uses the 2nd LSB of the first selected pixel value to refer to the index of the mismatch, it modifies the first pixel value differently from the second one in the selected pair of pixels. The embedding algorithm could be formulated in two separate forms:

$$p_s^{(2i)} = \begin{cases} p_c^{(2i)} + 2 & , \text{if } b_{2i} = LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i)} - 2 & , \text{if } b_{2i} \neq LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} = LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} + 1 & , \text{if } b_{2i} = \left[LSB\left(p_c^{(2i)}\right) = 0\right] \text{AND } b_{2i+1} = LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ & \text{OR } b_{2i} \neq \left[LSB\left(p_c^{(2i)}\right) = 0\right] \text{ AND } b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} - 1 & , \text{if } b_{2i} = \left[LSB\left(p_c^{(2i)}\right) = 1\right] \text{ AND } b_{2i+1} = LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ & \text{OR } b_{2i} \neq \left[LSB\left(p_c^{(2i)}\right) = 1\right] \text{ AND } b_{2i+1} \neq LSB\left(p_c^{(2i+1)}\right) \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ p_c^{(2i)} & , \text{otherwise} \end{cases} \quad \textbf{(4.6)}$$

$$p_s^{(2i+1)} = \begin{cases} p_c^{(2i+1)} + 1 \text{ , if } b_{2i} = LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} = \left[LSB\left(p_c^{(2i+1)}\right) = 0\right] \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ \quad \text{OR } b_{2i} \neq LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} \neq \left[LSB\left(p_c^{(2i+1)}\right) = 0\right] \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i+1)} - 1 \text{ , if } b_{2i} = LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} = \left[LSB\left(p_c^{(2i+1)}\right) = 1\right] \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 1 \\ \quad \text{OR } b_{2i} \neq LSB\left(p_c^{(2i)}\right) \text{ AND } b_{2i+1} \neq \left[LSB\left(p_c^{(2i+1)}\right) = 1\right] \text{ AND } 2^{nd}LSB\left(p_c^{(2i)}\right) = 0 \\ p_c^{(2i+1)} \quad \text{, Otherwise} \end{cases} \quad \textbf{(4.7)}$$

Where i is the index of the secret message bit where 0 ≤ i < (message length/2), $p_s^{(2i)}$ and $p_c^{(2i)}$ refer to the stego and clean pixel values respectively for the 2ith secret message bit embedding, and $p_s^{(2i+1)}$ and $p_c^{(2i+1)}$ again refer to the stego and clean pixel values used for embedding 2i+1th secret message bit, respectively.

The possible pixel value changes with SMLSB embedding could be simplified by separating the first $p_s^{(2i)}$ and the second $p_s^{(2i+1)}$ pixel values of the selected pair, as shown in Figure 4.5 and Figure 4.6.

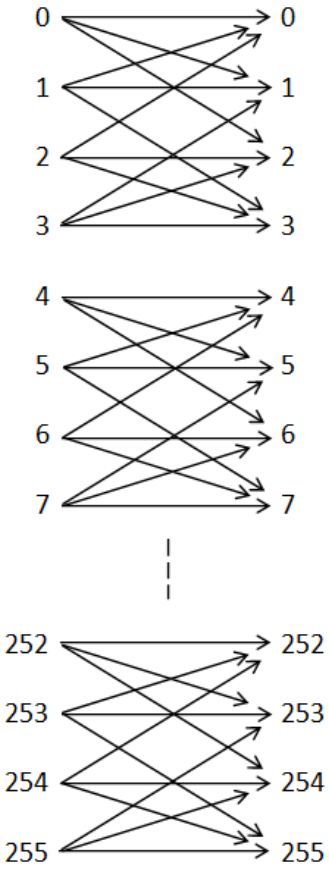


Figure 4.5: Possible pixel value transitions for $p_s^{(2i)}$ with SMLSB embedding

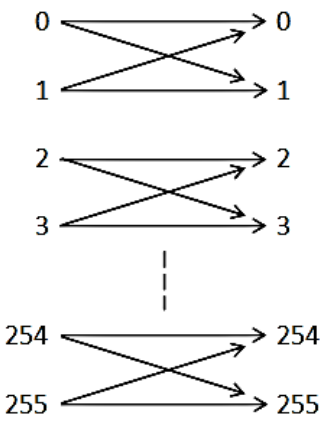


Figure 4.6: Possible pixel value transitions for $p_s^{(2i+1)}$ with SMLSB embedding

As could be noted from Figure 4.5 and Figure 4.6, the pixel value transitions of $\mathrm{p}_\mathrm{s}^{(2\mathrm{i}+1)}$ are like LSB replacement. While $\mathrm{p}_\mathrm{s}^{(2\mathrm{i})}$ is more complicated and has more transitions between clean and stego pixel values.

To analyse the impact of the SMLSB embedding on the intensity histogram, again we consider an embedding rate of p. Since the secret message is considered as a random sequence of 0 and 1, based on the fact that it will be close to its encrypted version (Chandramouli & Memon, 2001), equal probabilities should be considered for match/mismatch cases. Hence, for each case of $(\mathrm{MM}, \mathrm{M}\overline{\mathrm{M}}, \overline{\mathrm{M}}\mathrm{M}, \overline{\mathrm{MM}})$ the probability of occurrence would be 0.25.

For MM and $\overline{\mathrm{MM}}$, the embedding process will change one of the two selected pixel values according to the 2nd LSB of the $\mathrm{p}_\mathrm{c}^{(2\mathrm{i})}$ to get either $\mathrm{M}\overline{\mathrm{M}}$ or $\overline{\mathrm{M}}\mathrm{M}$. The change will be -1 or +1 for the odd and the even pixel values, respectively. So, $(\mathrm{p}/4)$ of the pixel values will be modified by adding or subtracting 1 according to their values, even or odd values respectively.

However, for $\mathrm{M}\overline{\mathrm{M}}$ and $\overline{\mathrm{M}}\mathrm{M}$ there is a probability of having 50% of the 2nd LSB of the first pixel value, $\mathrm{p}_\mathrm{c}^{(2\mathrm{i})}$, being the desired value, which needs no change. The other 50% will be modified by flipping the 2nd LSB of the $\mathrm{p}_\mathrm{c}^{(2\mathrm{i})}$ only. In other words, $(\mathrm{p}/8)$ of the pixel values will either increase or decrease by 2 according to their 2nd LSB value. Hence, the remaining $(1 - 3\mathrm{p}/8)$ pixel values will stay unchanged after embedding the secret message with the embedding rate of p, which means that embedding each message bit needs 0.375 pixel values to be changed. This ENMPP, 0.375, is better than LSB replacement and LSB matching, which are 0.5 pixels per message bit.

Hence, it improves the embedding efficiency from 2 to 8/3 bits per embedding change. The intensity histogram of the stego image could be estimated by the following:

$$h_s(n) = \left(1 - \frac{3p}{8}\right) h_c(n) + \frac{p}{8} \begin{cases} h_c(n+2) & \text{, if } 2^{nd} \text{ LSB}(n) = 0 \\ h_c(n-2) & \text{, if } 2^{nd} \text{ LSB}(n) = 1 \end{cases} + \frac{p}{4} \begin{cases} h_c(n+1) & \text{, n is even} \\ h_c(n-1) & \text{, n is odd} \end{cases} \quad \textbf{(4.8)}$$

Where n is again the grey-scale level valued between 0 and 255. Both $h_s(n)$ and $h_c(n)$ refer to the number of pixels in the stego and clean image respectively with the grey-scale value of n.

As only $(p/4)$ of the pixel values are modified like LSB replacement, it is expected that the probability of detection will be effectively reduced with LSB replacement and matching steganalysis methods, based on the dissimilarity in pixel value transitions and its influence on the intensity histogram after embedding.

The bit-level ENMPP for the proposed method could be calculated based on the match/mismatch cases, in which equal probabilities are considered:

$$\text{bit} - \text{level ENMPP} = \frac{\sum(\text{Probability of having each case} \times \text{no.of modified bits})}{2} \quad \textbf{(4.9)}$$

$$\text{bit} - \text{level ENMPP} = \frac{P_r(MM) \times 1 + P_r(M\overline{M}) \times 0.5 + P_r(\overline{M}M) \times 0.5 + P_r(\overline{MM}) \times 1}{2}$$

$$\text{bit} - \text{level ENMPP} = \frac{0.25 \times 1 + 0.25 \times 0.5 + 0.25 \times 0.5 + 0.25 \times 1}{2}$$

$$\text{bit} - \text{level ENMPP} = \frac{0.75}{2} = 0.375 \quad \text{bits per message bit}$$

The bit-level ENMPP is divided by two, as it embeds two bits of the secret message at a time. In this case the overall bit-level ENMPP for the proposed method will be 0.375 bits per message bit. Hence, the proposed method will result in fewer bit-level changes to the cover image after embedding the same amount of secret message. Table 4.2 shows the significant results from analysing LSB replacement, LSB matching and the proposed method.

Table 4.2: Analysis results of LSB replacement, LSB matching, and SMLSB

Embedding method	Stego noise probability	Embedding efficiency	ENMPP	Bit-level ENMPP
LSB replacement	$1-\frac{p}{2}$	2	0.5	0.5
LSB matching	$1-\frac{p}{2}$	2	0.5	≥ 0.5
SMLSB	$1-\frac{3p}{8}$	2.666	0.375	0.375

4.6.2 Experimental Results

To make the experimental results more reliable, two different sets of images are considered. The first set is 3000 images from ASIRRA (Animal Species Image Recognition for Restricting Access) public corpus pet images from the Microsoft Research website (Douceur, Elson, & Howell), which are random with different sizes, compression rates and texture etc. The other group is a set of 3000 never-compressed images from the Multimedia Forensics Group image database of Sam Houston State University ("Never-compressed image database,"). Both sets are used after conversion into grey-scale images.

To check the efficiency of the proposed embedding method, both detection methods are considered: the LSB replacement and the LSB matching steganalysis methods. In all experiments, streams of pseudo-random bits are considered as a secret message due to the fact that it will have all statistical properties of an encrypted version of the secret message according to (Westfeld & Pfitzmann, 2000). Also, to eliminate the effect of choosing the embedding place (random or sequential embedding), the embedding rate of one-bit per pixel (i.e. the images' total capacity) is considered. The proposed method is tested against both LSB replacement and matching steganalysis methods, as shown in the following sections.

4.6.2.1 SMLSB Against the Steganalysis Methods of LSB Replacement

There are many methods for detecting LSB replacement steganography in the literature; this paper considers two structural steganalysis methods, SP analysis (Dumitrescu et al., 2002) and Weighted Stego (WS) (Andrew D Ker & Böhme, 2008). These detection methods were chosen based on their accuracy in detection and the size estimation of the secret message. As mentioned earlier, for each case the image is loaded with the maximum capacity of the random secret message twice; one with LSB replacement and the other with SMLSB embedding.

The experimental results show that the proposed method effectively reduces the probability of detection for both LSB replacement detection methods over both sets of images compared to LSB replacement embedding method, as shown in Table 4.3.

Table 4.3: The overall reduction rates in probability of detection for SMLSB (in comparison to LSB).

Image set (3000) images	Detection method	The overall reduction in probability of detection
ASIRRA	WS	46.5%
Uncompressed	WS	48.4%
ASIRRA	SP	30.9%
Uncompressed	SP	39.8%

This reduction in the probability of detection has two advantages: the stego image would be considered as clean by the detector when the threshold value is a bit high or when the embedding rate is low; and the proposed embedding method results in a wrong estimation of the message length by the detection methods, therefore recovering the attributes of the secret message would be much more challenging, because the investigation starts after the estimation of the secret message as a first-level requirement of steganalysis.

Also, there is a noticeable reduction in probability of detection by the proposed method for the threshold value that suits the detection of LSB replacement, as shown in Figure 4.7 to Figure 4.10. This implies that the proposed method is much better than the LSB replacement in terms of undetectability, which is the most important property of steganography.

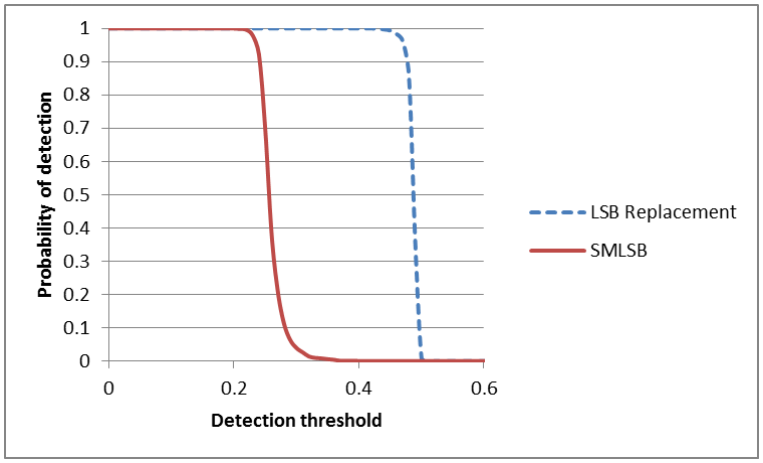


Figure 4.7: The probability of detection vs. detection threshold for ASIRRA images with WS

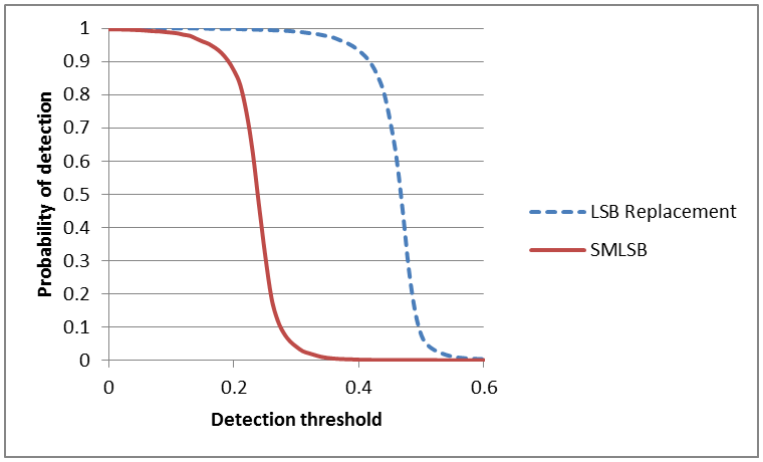


Figure 4.8: The probability of detection vs. detection threshold for uncompressed images with WS

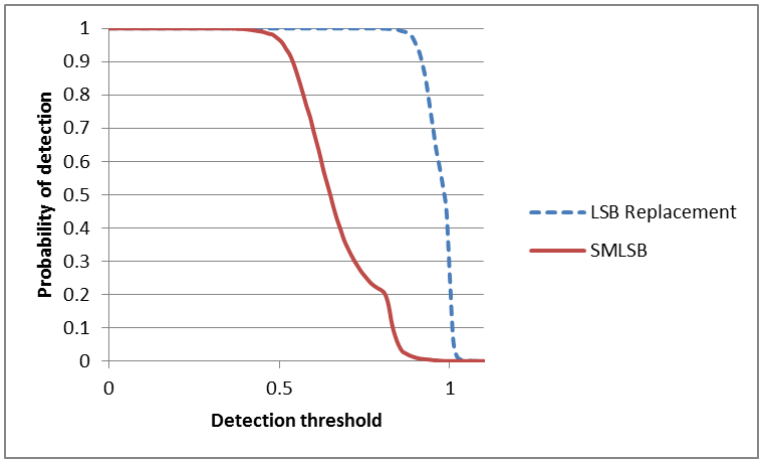


Figure 4.9: The probability of detection vs. detection threshold for ASIRRA images with SP

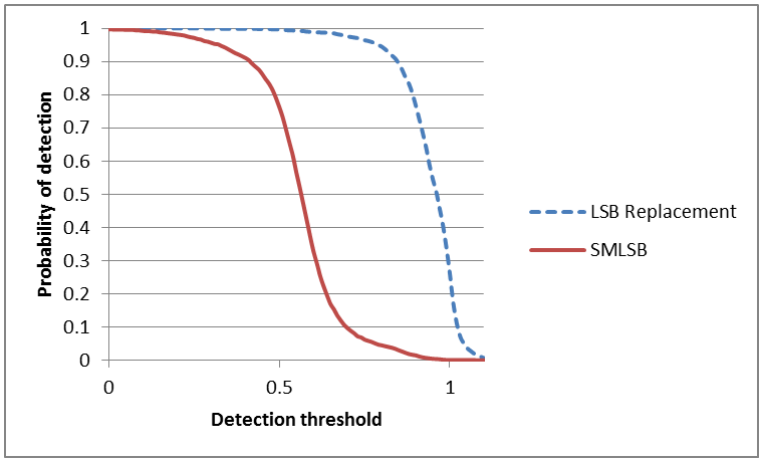


Figure 4.10: The probability of detection vs. detection threshold for uncompressed images with SP

As could be noticed, for the uncompressed set of images the probability of detection decreases sharply from 1 to, 0 which is not the case for uncompressed image set. Hence, using uncompressed images are better to be used for embedding.

4.6.2.2 *SMLSB Against the Steganalysis Methods of LSB Matching*

As mentioned earlier, there are two main categories of LSB matching steganalysis methods. One detection method in each category is used for testing the proposed embedding method. For the centre of mass of the histogram characteristic function (HCF-COM) the method of (A. D. Ker, 2005b) is used, and for the amplitude of local extrema the method proposed by (J. Zhang et al., 2007) is used.

Since there is no specific value to represent the best threshold for these two methods, the calculated values are shown in the graph. Also, due to the difficulty of showing all 3000 values separately in a clear manner on a single graph, the average of 30 values are taken to make one point on the graph. In this case for the entire 3000 images there are only 100 points, as shown in Figure 4.11 to Figure 4.14.

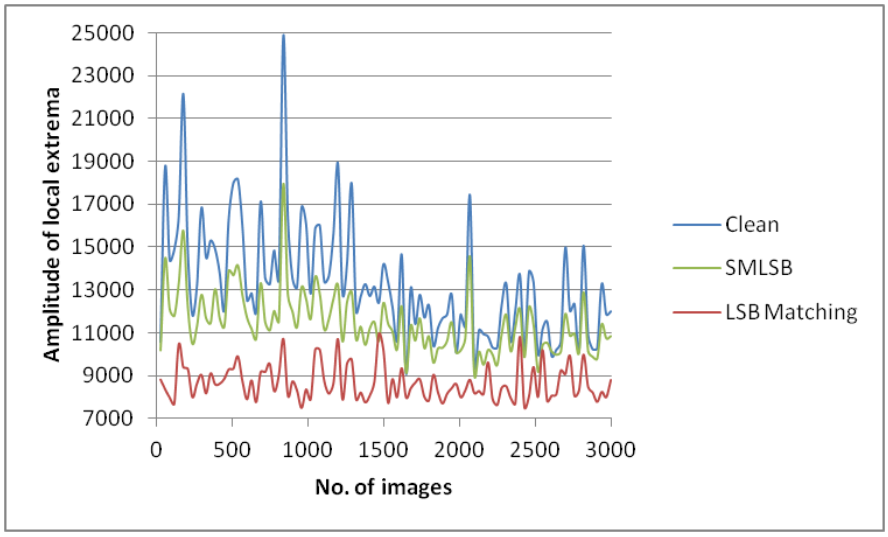


Figure 4.11: ALE values for clean, SMLSB, and LSB matching for ASIRRA images

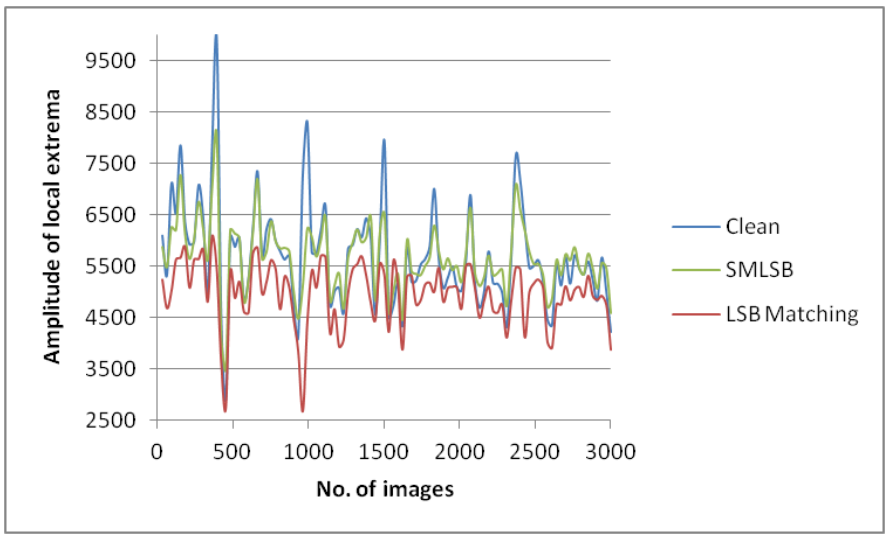


Figure 4.12: ALE values for clean, SMLSB, and LSB matching for uncompressed images

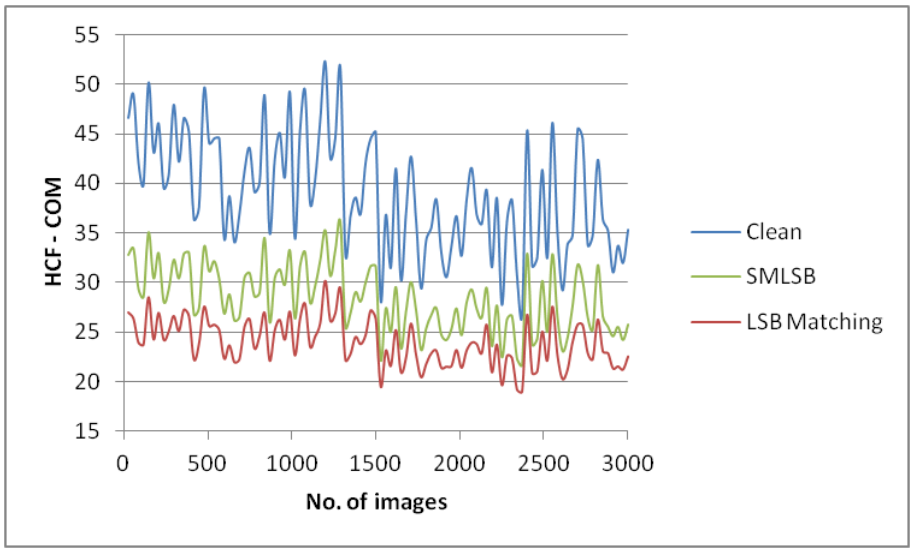


Figure 4.13: HCF-COM values for clean, SMLSB, and LSB matching for ASIRRA images

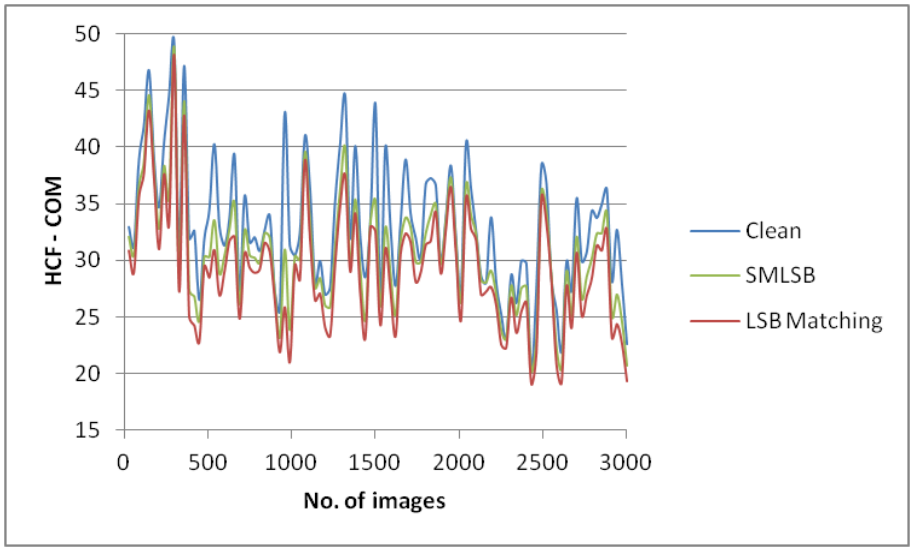


Figure 4.14: HCF-COM values for clean, SMLSB, and LSB matching for uncompressed images

It can be seen that the proposed method (SMLSB embedding) gives better results by having the higher ALE and HCF-COM values than the LSB matching embedding. Thus the ALE and HCF-COM values for the proposed method are closer to the clean image values, which leads to a lower probability of detection by the LSB matching steganalysis methods.

The ALE detects hidden content based on the amplitude of local extrema, so it is expected to perform better on uncompressed images. Moreover, due to changing the second-LSB of some pixel values by SMLSB, some stego images appear to have higher ALE values than the clean ones. For never-compressed images, 58% of the stego images had higher values than the clean ones. However, for the compressed images the rate was only 3%. Table 4.4, shows the better than clean results of stego images.

Table 4.4: The overall better results of stego images than clean ones

Image set	Detection method	The overall better detection values than clean images
ASIRRA	ALE	3%
Uncompressed	ALE	58%
ASIRRA	HCF-COM	0%
Uncompressed	HCF-COM	1%

The proposed method, SMLSB, outperforms both LSB matching and LSB matching revisited (Mielikainen, 2006) embedding methods in terms of detection. Figure 4.15 to Figure 4.18 show the ROC graphs for each group of images with the two selected detection methods. It can be seen from Figure 4.15 and Figure 4.16 that the ALE based steganalysis method is no more than a random classifier for the stego images embedded with the proposed method (SMLSB). Also, the performance of the HCF-COM based steganalysis method is considerably reduced by applying the proposed embedding method, as shown in Figure 4.17 and Figure 4.18.

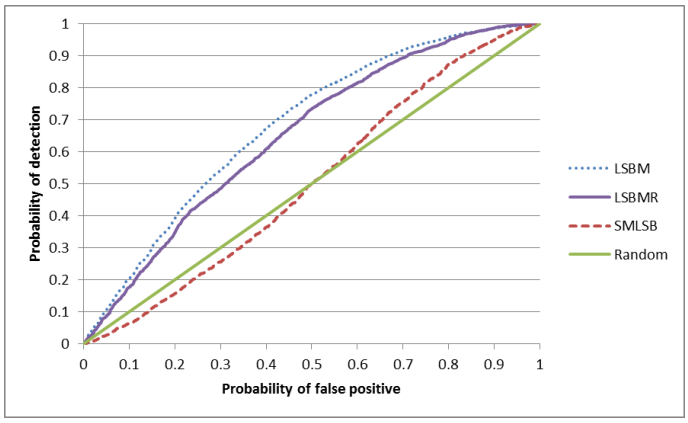


Figure 4.15: ROC graph of ALE steganalysis for LSB matching, LSB matching revisited, and SMLSB for ASIRRA images

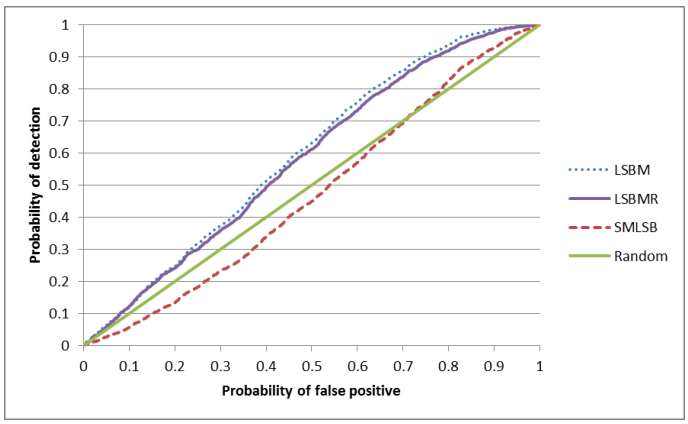


Figure 4.16: ROC graph of ALE steganalysis for LSB matching, LSB matching revisited, and SMLSB for uncompressed images

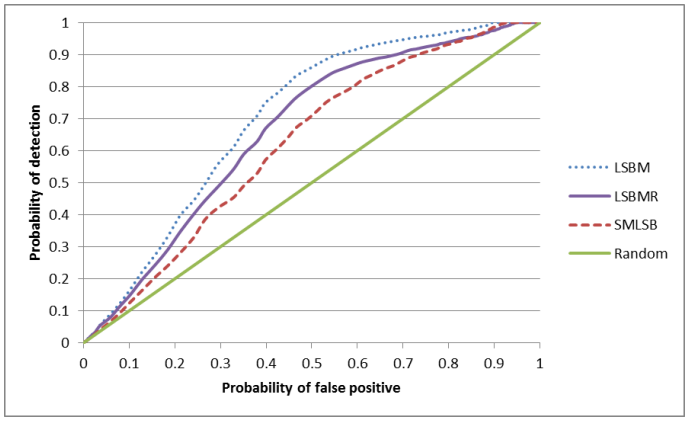


Figure 4.17: ROC graph of HCF-COM steganalysis for LSB matching, LSB matching revisited, and SMLSB for ASIRRA images

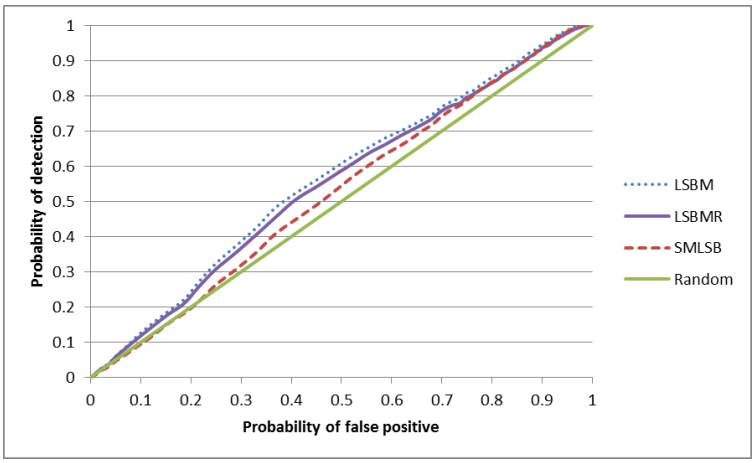


Figure 4.18: ROC graph of HCF-COM steganalysis for LSB matching, LSB matching revisited, and SMLSB for uncompressed images

Again, it could be noticed from the ROC graphs, the uncompressed images are less detectable than the compressed images. So, using the uncompressed images would give lower probability of detection.

Like any other steganography methods, the SMLSB cannot avoid all limitations and cannot totally defeat the detection methods. As noticed from Table 4.3 and Figure 4.7 to Figure 4.18, it is not possible to entirely avoid the detection. Also, there is another weakness regarding the image quality measurement PSNR between the cover and a stego image. The proposed method results in a slightly lower PSNR compared to LSB replacement, LSB matching and LSB matching revisited methods, which is still imperceptible and very far from the lower limit value of PSNR (38 dB) according to previous studies (Petitcolas & Anderson, 1999; K. Zhang, Gao, & Bao, 2009).

Table 4.5 shows the PSNR values for some standard images, shown in Figure 4.19, after embedding random binary streams with a maximum capacity using different embedding methods.

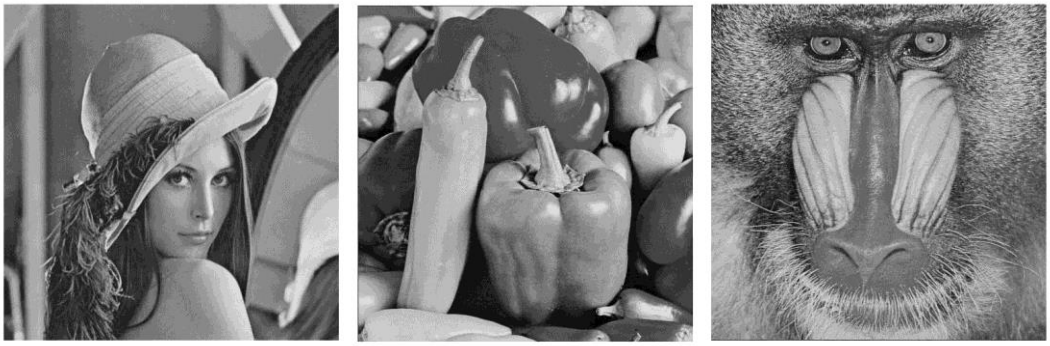


Figure 4.19: Three standard images used in steganography

Table 4.5: PSNR values in dB vs. embedding methods.

Images	LSB Replacement	LSB Matching	LSB Matching Revisited	SMLSB
Lenna	50.88	50.88	52.13	49.12
Pepper	50.17	50.17	51.41	48.42
Baboon	50.28	50.28	51.53	48.52

The same experiment is done for both sets of the compressed and uncompressed images, on average the PSNR of the proposed method was 1.75 dB less than the other LSB embedding methods.

4.6.3 Extraction Process of SMLSB

The extraction process could be simplified as follows. Let $s_1 s_2$ denote the least significant bits of the first and second selected pixel values respectively. The process considers the 2nd LSB of the first pixel value in the pair of pixels. If its value is 0, then the LSBs of the pair of pixels would be extracted in the form of $\bar{s_1} s_2$ as two secret message bits, since in this case, the mismatched LSB is in the first pixel value. If, on the other hand, it was 1, then it takes $s_1 \bar{s_2}$ as extracted message bits as the LSB of the second pixel value is mismatched. Table 4.6 shows all the different cases of the extraction process. Table 4.7 shows some examples of message bits extracted from stego pixel values.

Table 4.6: The extraction process

The stego images pixel pair	Extracted message bits
$xxxxxx0s_1$ $xxxxxxxs_2$	$\overline{s_1}s_2$
$xxxxxx1s_1$ $xxxxxxxs_2$	$s_1\overline{s_2}$

Table 4.7: Examples of SMLSB extraction process

The stego images pixel pair	Extracted message bits
xxxxxx01 xxxxxxx1	01
xxxxxx00 xxxxxxx1	11
xxxxxx11 xxxxxxx1	10
xxxxxx10 xxxxxxx1	00

4.7 Two Least Significant Bits Steganography (2LSB)

As LSB steganography in images became the most widely used embedding method, and is easily detectable by the current steganalysis methods, extensions to LSB steganography received great attention from steganographers (Luo et al., 2012; Yang et al., 2008; Xiaoyi Yu & Babaguchi, 2008; X. Yu et al., 2005). However, as claimed by (B. Li et al., 2011), embedding in multiple bit-planes may reduce the perceptual quality of the stego image, especially when high bit-planes are involved without considering the local property. The two least significant bit steganography has higher capacity than LSB embedding methods, is still easy to implement, visually imperceptible and results in more complex changes to the cover image pixel values, which would be harder to detect. The genuine superiority of 2LSB embedding to LSB embedding has been experimentally verified by (Andrew D Ker, 2007c).

Embedding in two least significant bits has been divided into two main categories, excluding the random selection of bit positions that could be applied in both cases. The 2LSB replacement directly replaces the 2LSB of the cover image's pixel value with 2-bits of the secret message. Independent 2LSB, known as I2LSB, replaces the 2LSB of the cover pixel values independently. For

instance, it can start with replacing the second-LSB of the pixel values with the secret message, then the first-LSB of the pixel values or vice versa. These methods of 2LSB embedding are clearly defined by (Andrew D Ker, 2007c) and (K. Zhang et al., 2009).

Many methods, like LSB matching, have been proposed to reduce the probability of detection in LSB embedding compared to LSB replacement. However, to the best of our knowledge, no method has been proposed in relation to 2LSB replacement to reduce the probability of detection or overcome the detection methods of 2LSB steganography. Enhancing 2LSB replacement is more complex than LSB, as there are four different cases of Match/Mismatch, shown in Table 4.8, which might be the reason for the lack of better methods for 2LSB embedding.

Table 4.8: Matching cases for LSB and 2LSB embedding

Embedding method	**Cases of Match/Mismatch**	
LSB	Match	
	Mismatch	
2LSB	Match	Match
	Match	Mismatch
	Mismatch	Match
	Mismatch	Mismatch

Detecting 2LSB embedding is much harder than detecting LSB embedding due to the complex changes in the cover image pixel values. A limited number of detection methods have been proposed to detect multiple least significant bits embedding using different concepts (Luo et al., 2012; Yang et al., 2008; Xiaoyi Yu & Babaguchi, 2008; X. Yu et al., 2005), as explained in more detail in chapter five, but they are not specific to 2LSB embedding and are expected to be less accurate than specific ones. Also explained in more detail in chapter five, some steganalysis methods have been proposed to detect 2LSB embedding (Andrew D Ker, 2007c; Luo, Wang, Yang, & Liu, 2006; Niu et al., 2009; K. Zhang et al., 2009). The method proposed in (Niu et al., 2009) is the most recent and accurate method in the literature, as claimed by the author and compared to the method proposed by Ker in (Andrew D Ker, 2007c). The method proposed by (Niu et al., 2009) constructs a weighted stego image and estimates the message length based on the least square method, which could be considered as a fast method of detection with high accuracy. Based on that, this method is selected as a detector to assess our proposed method of 2LSB embedding.

4.8 Improving the Embedding Efficiency and Undetectability of 2LSB

Both LSB and 2LSB replacement are weak, as they can be readily detected by many methods reported in the literature, as explained in previous sections. Improving these embedding methods became a dedicated field of research interest warranting more attention.

In this part of the research, a new embedding method is proposed for 2LSB steganography that makes fewer changes to the cover image with a lower probability of detection for the same amount of data compared to 2LSB replacement. The improvements of the new method are shown and proven in both theoretical and practical aspects.

4.8.1 Analysis of 2LSB Replacement

2LSB replacement data embedding technique is superior to LSB replacement, as discussed in previous sections. Analysing the 2LSB replacement would be very useful to specify its weak points, enhance its embedding efficiency and probability of detection. The analysis involves the embedding process, with its embedding efficiency, the changes to the intensity histogram, and the bit-level ENMPP for each pair of the secret message bits.

As explained in previous sections, there are two types of 2LSB steganography. The first method, 2LSB, directly replaces both least significant bits of the selected pixel value with 2-bits of the secret message. The second method, known as I2LSB, independently replaces the 2LSB of the selected pixel value with 2-bits of the secret message. Both embedding methods result in adding or subtracting the pixel value by 1, 2 or 3, according to the index of mismatch between the 2LSBs of the selected pixel value and 2-bits of the secret message. However, in the case of matching both 2LSBs of the selected pixel value with 2-bits of the secret message, the pixel value will stay unmodified. Figure 4.20, shows these value transitions between clean and stego pixel values.

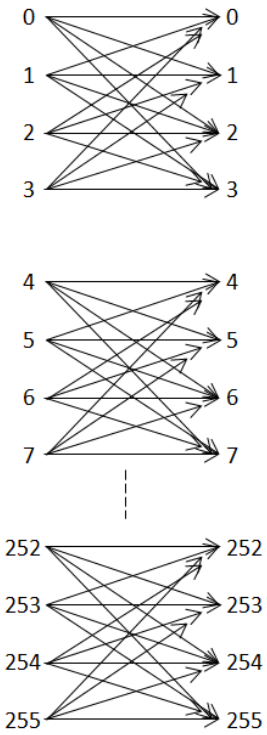


Figure 4.20: Possible transitions with I2LSB and 2LSB replacement

The embedding algorithm of 2LSB steganography could be formally represented as follows:

$$P_s = \begin{cases} P_c + 3 \;\; , if \; b_1 \neq [LSB(P_c) = 0] \; AND \; b_2 \neq [2^{nd} LSB(P_c) = 0] \\ P_c + 2 \;\; , if \; b_1 = LSB(P_c) \; AND \; b_2 \neq [2^{nd} LSB(P_c) = 0] \\ P_c + 1 \;\; , if \; b_1 \neq [LSB(P_c) = 0] \; AND \; b_2 = 2^{nd} LSB(P_c) \\ P_c - 1 \;\; , if \; b_1 \neq [LSB(P_c) = 1] \; AND \; b_2 = 2^{nd} LSB(P_c) \\ P_c - 2 \;\; , if \; b_1 = LSB(P_c) \; AND \; b_2 \neq [2^{nd} LSB(P_c) = 1] \\ P_c - 3 \;\; , if \; b_1 \neq [LSB(P_c) = 1] \; AND \; b_2 \neq [2^{nd} LSB(P_c) = 1] \\ P_c \;\; , Otherwise \end{cases} \quad \textbf{(4.10)}$$

Where P_s and P_c are the pixel values of stego and clean images respectively, and b1 and b2 represent the first and the second desired bits of the secret message to be hidden.

The intensity histogram of the cover image is directly affected by the embedding rate. According to the analysis done by (Andrew D Ker, 2007c), the probability of having Match/Mismatch cases between 2LSBs of the selected pixel value and the 2-bits of the secret message are not equal. Ker has represented those cases in the form of average distortion (0, ±1, ±2, ±3), as shown in Table 4.9.

Table 4.9: The stego noise probability for the methods of embedding in two LSBs

Embedding methods	Payload (bits)	Distortion			
		0	**±1**	**±2**	**±3**
2LSB	2Np	$1-\frac{3p}{4}$	$\frac{3p}{16}$	$\frac{p}{8}$	$\frac{p}{16}$
I2LSB	2Np	$1-p+\frac{p^2}{4}$	$\frac{3p}{4}+\frac{p^2}{16}$	$\frac{p}{4}-\frac{p^2}{8}$	$\frac{p^2}{16}$

Hence, the intensity histogram of the stego image could be estimated as follows:

$$h_s(n)=\left(1-\tfrac{3p}{4}\right)h_c(n)+\tfrac{3p}{16}[h_c(n+1)+h_c(n-1)]+\tfrac{p}{8}[h_c(n+2)+h_c(n-2)]+\tfrac{p}{16}[h_c(n+3)+h_c(n-3)] \qquad \textbf{(4.11)}$$

Where $h_s(n)$ and $h_c(n)$ represent the number of pixels in the stego and clean images with the grey-scale value of n respectively.

As could be noted from Table 4.9, the independent 2LSB (I2LSB) changes more pixels than 2LSB replacement for the same embedding rate. So, the embedding efficiency of the 2LSB would be better (higher) than the embedding efficiency of I2LSB. Hence, the 2LSB replacement is considered for analysis and comparison with the proposed method.

Even if equal probabilities of having Match/Mismatch $(M/\bar{M})$ cases are considered, the stego noise probability would be the same. Based on the fact that there are four cases, each case will have the probability of 25%, as shown in Table 4.10.

Table 4.10: The equal probability of Match/Mismatch cases in 2LSB steganography

Match/Mismatch cases	Probability of occurrences
MM	25%
$M\bar{M}$	25%
$\bar{M}M$	25%
$\bar{M}\bar{M}$	25%

Hence, the intensity histogram of the stego image could be estimated by the following formula, assuming the embedding rate of p:

$$h_s(n) = \left(1 - \frac{3p}{4}\right) h_c(n) + \frac{p}{8}[h_c(n+1) + h_c(n-1)] + \frac{p}{8}[h_c(n+2) + h_c(n-2)] + \frac{p}{8}[h_c(n+3) + h_c(n-3)] \quad \textbf{(4.12)}$$

Again, $h_s(n)$ and $h_c(n)$ represent the number of pixels in the stego and clean images with the grey-scale value of n respectively. In both cases, the expected number of modifications per pixel (ENMPP) is 0.75. Hence, the embedding efficiency of 2LSB replacement is 2.666 bits per embedding change.

The bit-level ENMPP of the 2LSB replacement is also another important perspective to be analysed, especially for detection methods that rely on the binary similarity measures between low bit-planes (Avcibaş et al., 2005). The bit-level ENMPP of the 2LSB replacement could be estimated by multiplying the probability of different Match/Mismatch cases by the number of bits required to change.

For the probabilities considered by (Andrew D Ker, 2007c), the bit-level ENMPP would be estimated as follows:

$$bit - level\ ENMPP = \sum(Probability\ of\ having\ each\ case \times no.\ of\ modified\ bits) \quad \textbf{(4.13)}$$

$$= Pr(MM) \times 0 + Pr(\bar{M}\bar{M}) \times 2 + Pr(M\bar{M}) \times 1 + Pr(\bar{M}M) \times 1$$

$$bit - level\ ENMPP = (0.25 \times 0) + (0.125 \times 2) + (0.375 \times 1) + (0.25 \times 1)$$

$$= 0 + 0.25 + 0.375 + 0.25$$

$$= 0.875\ bits/\ two\ message\ bits$$

So, the overall bit-level ENMPP of 2LSB replacement is 0.875 bits for each pair of the secret bits, having different probabilities of Match/ Mismatch cases.

However, if equal probabilities of having Match/ Mismatch cases are considered, this bit-level ENMPP would be even higher by equation 4.13, which is 1 bit per two message bits, as shown below:

$$bit - level\ ENMPP = Pr(MM) \times 0 + Pr(\bar{M}\bar{M}) \times 2 + Pr(M\bar{M}) \times 1 + Pr(\bar{M}M) \times 1$$

$$= (0.25 \times 0) + (0.25 \times 2) + (0.25 \times 1) + (0.25 \times 1)$$

$$= 1\ bit/\ two\ message\ bits$$

4.9 Single Mismatch 2LSB Steganography (SM2LSB)

Both methods of two LSB steganography, I2LSB and 2LSB replacement, are vulnerable to steganalysis attacks with a high probability of detection. Hence, the new proposed method, single mismatch 2LSB steganography (SM2LSB) is proposed to improve the embedding efficiency and undetectability of 2LSB replacement. The proposed method can be applied for the embedding rate of 1, without restrictions on the features of the cover image.

Hence, the SM2LSB is a non-adaptive embedding method that can be applied on any pixel values of the cover image without restricting the image capacity. It is also intended to reduce the number of bit changes required for embedding the same amount of the secret message embedded with 2LSB replacement.

The proposed method considers a single mismatch between the 2LSB of the selected pixel value and 2-bits of the secret message and uses the third LSB as a pointer to the index of the mismatch. Based on the Match/ Mismatch cases, there are four possible combinations, as shown in Table 4.8, where M and $\bar{M}$ denote the match and mismatch between the secret message bit and the bit-plane of the pixel value.

In the case of MM and $\bar{M}\bar{M}$, the embedding process changes one of the 2LSB of the selected pixel value to get $M\bar{M}$ or $\bar{M}M$ according to the binary value of the third least significant bit. If the third LSB of the selected pixel value was 0, then the mismatch would be in the first-LSB and the result after embedding would be ($M\bar{M}$). If it was 1, the mismatch would be in the second-LSB and the result after embedding would be ($\bar{M}M$), as shown in Table 4.11.

Table 4.11: The relation between third LSB and single mismatch

Third LSB	Match/ Mismatch cases
0	$M\bar{M}$
1	$\bar{M}M$

For the other two cases ($M\bar{M}$ and $\bar{M}M$), the embedding method changes the third LSB of the selected pixel value according to the index of the mismatch. It sets the third LSB to 0 for $M\bar{M}$ and sets it to 1 for $\bar{M}M$ cases, as shown in Table 4.11. However, there is a probability of 50% that the third LSB already have a right index value and therefore no change would be done to the pixel value. Using the third LSB will affect the transparency of the embedding algorithm, but as shown in section 4.9.2, the stego images are still imperceptible. The embedding algorithm for a single pixel value and 2-bits of the secret message is shown in Figure 4.21.

Input: cover image pixel value x, two message bits m_i, m_{i+1}
Output: stego image pixel value y
$y = x$
if $2^{nd}\text{LSB}(x) = m_i \; AND \; LSB(x) = m_{i+1}$
  if $3^{rd} LSB(x) = 0$
    $LSB(y) = \overline{m_{i+1}}$
  else
    $2^{nd} LSB(y) = \overline{m_i}$
else if $2^{nd}\text{LSB}(x) \neq m_i \; AND \; LSB(x) \neq m_{i+1}$
  if $3^{rd} LSB(x) = 0$
    $2^{nd} LSB(y) = \overline{m_i}$
  else
    $LSB(y) = \overline{m_{i+1}}$
else if $2^{nd}\text{LSB}(x) = m_i \; AND \; LSB(x) \neq m_{i+1}$
  $3^{rd} LSB(y) = 0$
else
  $3^{rd} LSB(y) = 1$
end

Figure 4.21: Proposed embedding algorithm (SM2LSB) for 2-bits of the secret message

Table 4.12 shows some examples of embedding cases. As can be noted, for each embedding case there is a maximum of 1-bit change or no change to the three LSBs of the cover pixel values. Changes are highlighted by shading the cell.

Table 4.12: Examples of embedding pairs of message bits into cover image

Cover's 3LSB			**Secret message**		**Stego's 3LSB**		
C3	C2	C1	m2	m1	S3	S2	S1
0	0	0	0	0	0	0	1
1	1	1	1	1	1	0	1
0	0	1	1	0	0	1	1
1	1	0	0	1	1	1	1
0	1	0	1	1	0	1	0
1	0	1	0	0	0	0	1
1	1	1	0	1	1	1	1

4.9.1 Analysis of SM2LSB Embedding

The analysis of SM2LSB is done in three perspectives: the embedding process with its embedding efficiency, the changes to the intensity histogram, and the bit-level cost of change. The embedding process relies on the Match/ Mismatch cases between the 2LSBs of the selected pixel value and 2-bits of the secret message. Hence, the transitions of the pixel values are different from 2LSB replacement, as shown in Figure 4.22.

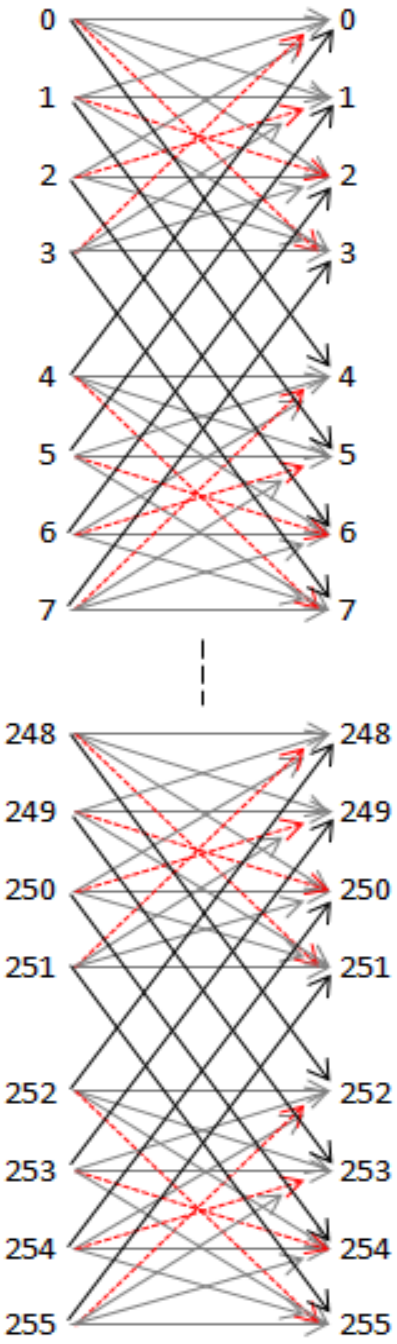


Figure 4.22: Possible pixel value transitions with SM2LSB embedding

As can be seen in Figure 4.22, the SM2LSB eliminates the value transitions caused by changing both LSBs, indicated by dashed red arrows. It adds other pixel value transitions of (±4), caused by changing the third LSB, drawn with bold arrows. This difference in pixel value transitions would be one of the reasons of reducing the probability of detection, as shown in section 4.9.2.

The embedding algorithm of the SM2LSB could be formally represented as follows:

$$P_s = \begin{cases} P_c + 4 \ \ ,if\ b_1 = LSB(P_c)\ AND\ b_2 \neq 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 0 \\ P_c + 2 \ \ ,if\ b_1 = LSB(P_c)\ AND\ b_2 = 2^{nd}LSB(P_c) = 0\ AND\ 3^{rd}LSB(P_c) = 1 \\ \quad OR\ b_1 \neq LSB(P_c)\ AND\ b_2 \neq [2^{nd}LSB(P_c) = 0]\ AND\ 3^{rd}LSB(P_c) = 0 \\ P_c + 1 \ \ ,if\ b_1 = LSB(P_c) = 0\ AND\ b_2 = 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 0 \\ \quad OR\ b_1 \neq [LSB(P_c) = 0]\ AND\ b_2 \neq 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 1 \\ P_c - 1 \ \ ,if\ b_1 = LSB(P_c) = 1\ AND\ b_2 = 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 0 \\ \quad OR\ b_1 \neq [LSB(P_c) = 1]\ AND\ b_2 \neq 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 1 \\ P_c - 2 \ \ ,if\ b_1 = LSB(P_c)\ AND\ b_2 = 2^{nd}LSB(P_c) = 1\ AND\ 3^{rd}LSB(P_c) = 1 \\ \quad OR\ b_1 \neq LSB(P_c)\ AND\ b_2 \neq [2^{nd}LSB(P_c) = 1]\ AND\ 3^{rd}LSB(P_c) = 0 \\ P_c - 4 \ \ ,if\ b_1 \neq LSB(P_c)\ AND\ b_2 = 2^{nd}LSB(P_c)\ AND\ 3^{rd}LSB(P_c) = 1 \\ P_c \qquad ,Otherwise \end{cases} \quad \textbf{(4.14)}$$

Analysing the influence of SM2LSB on the intensity histogram would be done for both equal and different probabilities of Match/ Mismatch cases. According to (Andrew D Ker, 2007c), the probabilities of having Match/ Mismatch cases are different, as shown in Table 4.13.

Table 4.13: The different probabilities of Match/ Mismatch cases for 2LSB steganography

Match/ Mismatch cases	Probability of occurrences
MM	$1 - \frac{3p}{4}$
$M\bar{M}$	$\frac{3p}{8}$
$\bar{M}M$	$\frac{p}{4}$
$\bar{M}\bar{M}$	$\frac{p}{8}$

Hence, the intensity histogram of the stego image could be estimated by the following formula:

$$h_s(n) = \left(1 - \tfrac{11p}{16}\right) h_c(n) + \tfrac{3p}{32}[h_c(n+1) + h_c(n-1) + h_c(n+2) + h_c(n-2)] + \tfrac{5p}{32}[h_c(n+4) + h_c(n-4)] \quad \textbf{(4.15)}$$

Where $h_s(n)$ and $h_c(n)$ are the number of pixels in stego and clean images with the grey-scale value of n.

The changes to both $M\bar{M}$ and $\bar{M}M$ cases are $\frac{5p}{16}$, but due to having the probability of 50% that the third LSB already contain the right index value, this rate is changed to $\frac{5p}{32}$. For the other two cases

MM and $\bar{M}\bar{M}$, the overall probability of pixel value changes would be $\frac{3p}{8}$. Hence, $\frac{3p}{32}$ for each case of change (±1 and ±2). As could be noticed, the rate of unmodified pixel values is higher than embedding with 2LSB replacement, which directly affects the probability of detection by the 2LSB steganalysis methods.

However, if an equal probability of Match/ Mismatch cases are considered, then each case ($MM, M\bar{M}, \bar{M}M, \bar{M}\bar{M}$) will have the probability of $\frac{p}{4}$ for the embedding rate of p. Then, the intensity histogram of the stego image would be estimated as follows:

$$h_s(n) = \left(1 - \frac{3p}{4}\right) h_c(n) + \frac{p}{8}[h_c(n+1) + h_c(n-1) + h_c(n+2) + h_c(n-2)] + \frac{p}{8}[h_c(n+4) + h_c(n-4)] \qquad \textbf{(4.16)}$$

Again the overall probability of having $M\bar{M}$ and $\bar{M}M$ cases are $\frac{p}{2}$, and there is a probability of 50% that the third LSB of the pixel value contains the desired value. Hence, the probability of changing pixel values of $M\bar{M}$ and $\bar{M}M$ would be $\frac{p}{4}$, or $\frac{p}{8}$ for each pixel value changes (±4). For other two cases MM and $\bar{M}\bar{M}$, the overall probability of pixel value changes is $\frac{p}{2}$, or $\frac{p}{8}$ for each pixel value change (±1 and ±2).

The proposed method, SM2LSB, is still better than 2LSB replacement in terms of bit-level ENMPP, as explained in the following paragraphs. The expected number of modifications per pixel, ENMPP, would be different for different probabilities of Match/ Mismatch cases. For the probabilities considered by (Andrew D Ker, 2007c), the ENMPP is 0.6875 for the proposed method, which is less than I2LSB and 2LSB replacement (0.75). However, if equal probabilities of having Match/ Mismatch cases are considered, the ENMPP for the proposed method is 0.75, which is equal to 2LSB replacement. Moreover, as it has different pixel value transitions from 2LSB replacement, it also gives less probability of detection, as explained in section 4.9.2.

The last aspect of the analysis is the bit-level ENMPP, which is also important, especially for steganalysis methods that rely on the binary similarity measures (Avcibaş et al., 2005). This method uses the seventh and eighth bit planes in an image to compute several binary similarity measures. The bit-level ENMPP is found by multiplying the probability of having Match/Mismatch cases by the number of bits needed to change.

For the probability of Match/ Mismatch cases considered by Ker (Andrew D Ker, 2007c), based on equation 4.13, the bit-level ENMPP could be estimated as follows:

$$bit - level\ ENMPP = Pr(MM) \times 1 + Pr(\bar{M}\bar{M}) \times 1 + Pr(M\bar{M}) \times 0.5 + Pr(\bar{M}M) \times 0.5$$

$$= (0.25 \times 1) + (0.125 \times 1) + (0.375 \times 0.5) + (0.25 \times 0.5)$$

$$= 0.25 + 0.125 + 0.1875 + 0.125$$

$$= 0.6875\ bits\ per\ two\ message\ bits$$

The cases with either match or mismatch are very clear, as it changes 1-bit to match its index in third LSB. For other cases, Match-Mismatch and Mismatch-Match, the proposed embedding method would look at the index of the mismatch and sets the third LSB of the pixel value accordingly: 0 if the mismatch was in first-LSB and 1 if it was in second-LSB. However, there is a probability of 50% that the third LSB contains the right index, which needs no change, and the other 50% needs only 1-bit to change, so in this case it needs only 0.5 bits to change for embedding 2-bits of the secret message.

The bit-level ENMPP for 2LSB replacement was 0.875, with equal probabilities of having Match/ Mismatch cases. Hence, the embedding process with SM2LSB reduces the number of modified bits (to 0.6875) for embedding the same amount of secret data compared with 2LSB replacement.

Moreover, if equal probability of having Match/ Mismatch cases is considered, by equation 4.13 the bit-level ENMPP is estimated as follows:

$$bit - level\ ENMPP = Pr(MM) \times 1 + Pr(\bar{M}\bar{M}) \times 1 + Pr(M\bar{M}) \times 0.5 + Pr(\bar{M}M) \times 0.5$$

$$= (0.25 \times 1) + (0.25 \times 1) + (0.25 \times 0.5) + (0.25 \times 0.5)$$

$$= 0.25 + 0.25 + 0.125 + 0.125$$

$$= 0.75\ bits\ per\ two\ message\ bits$$

This rate (0.75) is again less than the 2LSB replacement rate, which is 1. Hence, for both cases of equal and non-equal probabilities of Match/ Mismatch cases, the bit-level ENMPP for the embedding method is less than that for the 2LSB replacement. This is one of the advantages of the proposed method over 2LSB replacement.

The bit-level ENMPP is always equal to the normal ENMPP for the proposed embedding method. However, for 2LSB replacement, the bit-level ENMPP is always higher than the normal ENMPP. Table 4.14 shows the analysis results of both 2LSB replacement and the proposed method.

Table 4.14: Analysis results of 2LSB replacement and SM2LSB

Embedding method	Match/ Mismatch Probabilities	Stego noise probability	Embedding efficiency	ENMPP	Bit-level ENMPP
2LSB replacement	Different (Andrew D Ker, 2007c)	$1-\frac{3p}{4}$	2.666	0.75	0.875
2LSB replacement	Equal (0.25% each)	$1-\frac{3p}{4}$	2.666	0.75	1
SM2LSB	Different (Andrew D Ker, 2007c)	$1-\frac{11p}{16}$	2.909	0.6875	0.6875
SM2LSB	Equal (0.25% each)	$1-\frac{3p}{4}$	2.666	0.75	0.75

This shows theoretically that the proposed method causes fewer changes to the cover image and is expected to result in lower probability of detection for the same secret message embedded by 2LSB replacement. To confirm this practically, three standard images, shown in Figure 4.19, were chosen and embedded twice with a maximum capacity of random messages for 2LSB embedding; one with 2LSB replacement, and the other with SM2LSB. The stego images then analysed by the detection method proposed by (Niu et al., 2009). The SM2LSB embedding causes more distortions than 2LSB replacement, as shown by calculating the PSNR, which is still acceptable, as the changes could be noticed only when the PSNR value is less than 38dB according to (K. Zhang et al., 2009). Table 4.15 shows the distortion and the probability of detection in both cases. It can be seen that the proposed method has less probability of detection than 2LSB replacement by 44.6%.

Table 4.15: Probability of detection vs. distortion

Images	Probability of Detection		PSNR	
	2LSB	SM2LSB	2LSB	SM2LSB
Lenna	0.396	0.222	44.79	40.98
Pepper	0.407	0.220	44.79	40.97
Baboon	0.425	0.238	44.79	40.96

The same experiment is repeated for both sets of the compressed and uncompressed images, on average the PSNR of the SM2LSB was 3.8 dB less than the 2LSB replacement.

4.9.2 Experimental Results

To evaluate the proposed method, two sets of compressed and uncompressed images were used. The first set consists of 3000 random images from ASIRRA public corpus images, and 3000 never-compressed images from Multimedia Forensics Group image database of Sam Houston State University, as explained previously. Both image sets were used as cover images after converting them into grey-scale, without changing their dimensions. One of the latest and most accurate targeted 2LSB detection methods (Niu et al., 2009) was used to find the probability of detection. The cover images were then loaded with a random message, to be close to the encrypted version (Westfeld & Pfitzmann, 2000), with the length of maximum capacity for 2LSB embedding.

The embedding is done twice with the same random bit-stream of the secret message, one with 2LSB replacement and the other with SM2LSB. On average, for the 6000 images, the probability of detection is reduced by more than 44% compared to 2LSB replacement. This reduction results in a very high false negative rate for the same threshold that suits 2LSB replacement.

To visualise the difference in detection between 2LSB replacement and SM2LSB embedding, the true positive rates were taken in relation to the threshold of detection. As shown in Figure 4.23 and Figure 4.24, which show the scaled to the area of difference, the same true positive rate could be gained only when the threshold of detection is reduced by more than 44%, which implies increasing sensitivity of detection.

The only disadvantage of the proposed method is the additional distortion to the cover image, which makes the value of PSNR on average 3.8 dB less than the 2LSB replacement according to the experimental results on the set of 6000 images, which is still much more than the perceptible value - 38 dB (Petitcolas & Anderson, 1999; K. Zhang et al., 2009).

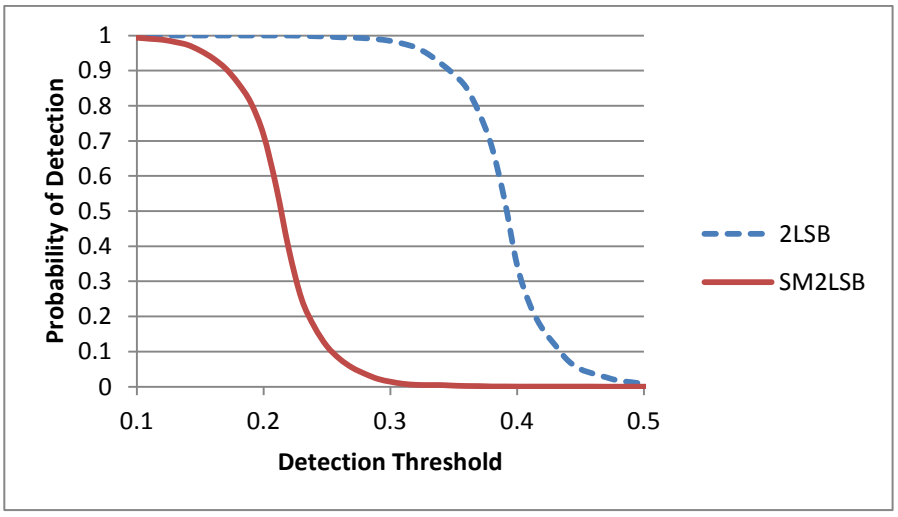


Figure 4.23: The probability of detection for SM2LSB and 2LSB replacement - uncompressed images

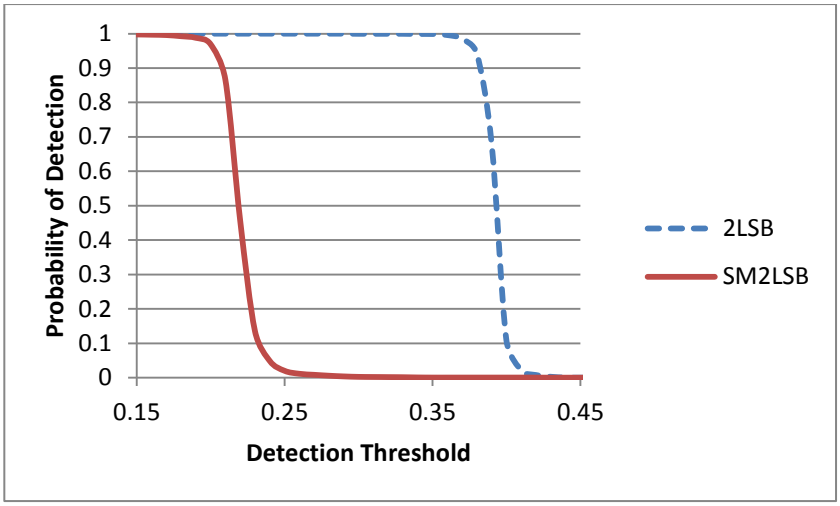


Figure 4.24: The probability of detection for SM2LSB and 2LSB replacement - ASIRRA images

Again, it could be noticed that the probability of detection for the compressed set of images sharply falls down which is not the case with uncompressed set of images. Hence, again uncompressed images are preferred to be used as a cover object.

4.9.3 Extraction Process of SM2LSB

The extraction process looks at the third LSB of the stego image's pixel value, if it is 1; the message would be the complement of the second-LSB and the first-LSB ($\bar{S}_2 S_1$), otherwise, if it is 0; then the secret message is ($S_2 \bar{S}_1$), as shown in Table 4.16.

Table 4.16: Extraction process

The stego's 3LSBs			**2-bits of the secret message**	
0	S_2	S_1	S_2	$\bar{S}_1$
1	S_2	S_1	$\bar{S}_2$	S_1

Table 4.17 shows some examples of the pixel values and the recovered 2-bits of the secret message.

Table 4.17: Examples of SM2LSB extraction process

The stego images pixel pair	Extracted message bits
xxxxx000	01
xxxxx100	10
xxxxx001	00
xxxxx101	11
xxxxx010	11
xxxxx110	00
xxxxx011	10
xxxxx111	01

4.10 Conclusion

In this chapter the proposed embedding method (single mismatch) was applied in two different types of embedding methods, LSB and 2LSB steganography. Both methods attempt to create a single mismatch cases.

The detection results of the proposed method showed that the proposed method (SMLSB) can effectively improve the embedding efficiency in comparison to LSB replacement and LSB matching from 2 to 8/3 and reduce the probability of detection with both LSB steganalysis (LSB replacement and LSB matching). It also leaves a higher rate of pixel values unchanged for embedding the same amount of secret messages compared with the two LSB steganography methods. Moreover, the proposed method outperforms the LSB matching revisited, which has the same embedding efficiency in terms of detection. Also, the new method, unlike the other methods discussed in section 4.5, can be applied to any pixel without skipping the saturated values (0 and 255).

Another important point is that all LSB embedding methods are analysed in detail, including SMLSB, and the cause of reducing the probability of detection is also highlighted. As could be noticed, the proposed method is very simple to implement, with no complex calculations, less bit-level ENMPP on the cover image, and no reduction in the embedding capacity compared to the other two LSB steganography methods, LSB replacement and LSB matching.

In addition to the lower probability of detection, the proposed method would result in a wrong estimation of the message length by any detection method. Consequently, it misguides the investigator by having insufficient information about the attributes of the hidden message.

Since it modifies some of the pixel values during the embedding process, reducing the probability of detection by LSB steganalysis methods is limited and the new method cannot totally avoid it.

Also, it results in slightly more distortion in comparison to LSB replacement and LSB matching methods.

The other proposed embedding method, SM2LSB, allows embedding the same amount of data with less change to the cover image. There is a probability of changing only 0.6875 pixel values of the cover image for each pair of the secret message, while this probability is 0.75 for 2LSB replacement. Hence, the proposed method has a higher embedding efficiency, which directly affects the probability of detection. On average, the proposed method reduces the probability of detection by more than 44% compared to embedding the same amount of the secret message with 2LSB replacement.

The chosen steganalysis method can maintain the detection accuracy only when the threshold value is about half the value of the 2LSB replacement detection threshold. As shown in the experimental results, the proposed method affected the performance of the detection and forced it to give a very low true positive rate for the same threshold that suits 2LSB replacement.

The proposed method also costs less bit-level changes than the 2LSB replacement for both probabilities of having Match/ Mismatch cases. For unequal probabilities, the bit-level ENMPP of the proposed method is 0.6875 bits, and 0.875 bits for 2LSB replacement. For equal probabilities, the bit-level ENMPP of the proposed method is 0.75 bits, and 1 bit for 2LSB replacement. This reduction in bit-level modification reduces the probability of detection by the steganalysis methods that rely on binary similarity measures between low bit-planes.

In addition to all previously mentioned advantages, the proposed method affects the investigation process of recovering the secret message by giving a shorter estimated message length than the actual length, as the recovery process relies on the estimated properties of the hidden message.

Finally, as in some cases the proposed method modifies the third bit of the pixel value, this results in a lower PSNR by 3.8 dB on average compared to 2LSB replacement. However, the PSNR values are still much higher than the perceptible value, which is 38 dB (Petitcolas & Anderson, 1999; K. Zhang et al., 2009).

CHAPTER 5: DETECTING THE 2LSB STEGANOGRAPHY VIA EXTENDED PAIRS OF VALUES

5.1 Introduction

Usually, multimedia digital objects are excellent media for steganography, as they have a high degree of redundancy (Chandramouli & Memon, 2001). Thus, digital images are one of the best and most commonly used digital media for this purpose. The most commonly used method of image steganography is the LSB embedding. The reason behind the interest in LSB steganography is that it is easy to implement, has a reasonable capacity, and is visually imperceptible. However, it could be easily detected due to the imbalance distortion on the intensity histogram of the image and producing 'Pairs of Values'. There are numerous studies on LSB steganography in images (Chutani & Goyal, 2012), and lots of methods have been proposed to detect the existence of embedded message. As a result, extensions to LSB steganography received great attention by steganographers, and nowadays there are a number of publicly available steganography tools that could be used for this purpose, for example SilentEye (Chorein, 2008), which allows the use of more than one LSB and different colour components (RGB; red, green, and blue) for embedding. One of the extensions of the LSB steganography method is 2LSB data embedding, which has even higher capacity than LSB method with more complicated changes on the intensity histogram of the cover image, making it harder to detect.

The reason behind considering targeted steganalysis method is that the universal steganalysis methods are less accurate (Kharrazi et al., 2006), it does not need to know what embedding method is used, and a number of detection techniques have to be deployed on the analysed image. It was also observed by previous studies (Avcibaş et al., 2005; Kharrazi, Sencar, & Memon, 2005) that the universal methods do not perform equally over all steganographic methods. Furthermore, the universal classifier needs to be trained using a set of sample cover and stego images, which is computationally expensive based on the type of classifier, the size of sample dataset and the differentiation of cover and stego images in the feature space. Additionally, the universal steganalysis methods may suffer from cover source mismatch, because it has to be trained with a certain sample set of cover and stego images (Pevný & Ker, 2013).

In this chapter, an EPoV analysis is presented to detect and estimate the amount of secret messages embedded with 2LSB replacement in digital images, based on chi-square and standard deviation attacks. In chi-square attack, the detection process is separated from the estimation of

the hidden message length, as it is the main requirement of any steganalysis method. Hence, the detection process can act as a discrete classifier, which classifies a given set of images into stego and clean classes. The method can accurately detect 2LSB replacement even when the message length is about 10% of the total capacity. It also reaches its best performance with an accuracy of higher than 0.96 and a true positive rate of more than 0.997 when the amount of data is 20% to 100% of the total capacity. However, the method puts no assumptions either on the image or the secret message, as it is tested with two sets of 3000 images, compressed and uncompressed, embedded with a random message for each case. This method of detection could also be used as an automated tool to analyse a bulk of images for hidden contents, which could be used by digital forensics analysts in their investigation process.

However, the standard deviation attack measures the amount of distortion in the stego image made by the embedding process of 2LSB replacement, which is directly proportional with the embedding rate. It is shown that it can accurately estimate the length of the hidden message and outperform the other methods of the targeted 2LSB steganalysis in the literature. The proposed method is also more consistent with the steganalysis methods reported by previous studies by giving the amount of difference to the expected clean image. According to the experimental results, based on analysing 3000 never-compressed images, the proposed method is more accurate than the current targeted 2LSB steganalysis methods for low embedding rates.

This chapter starts with a brief description of the principle of pairs of values with its analysis and the 2LSB steganography in digital images, then it explains the proposed new form of pairs of values with the 2LSB steganalysis methods in the literature. Subsequently, the chi-square and standard deviation methods of attacks are proposed, and their detection accuracy is experimentally tested.

5.2 Pairs of Values

The concept of PoV is obtained from the pixel value transitions of LSB replacement in digital images. The LSB replacement embeds the secret message by replacing the LSB of the cover image's pixel value with each bit of the secret message. Hence, the value transitions between the cover and stego images will be limited by having differences only in their LSB value. Moreover, adding one to even values and subtracting one from odd pixel values will produce pairs of values with grey-scale value of $2k$ and $2k+1$, where $0 \leq k \leq 127$, as shown in Figure 5.1. These pairs of values are used as a base of detection by (Westfeld & Pfitzmann, 2000) via pairs of values analysis.

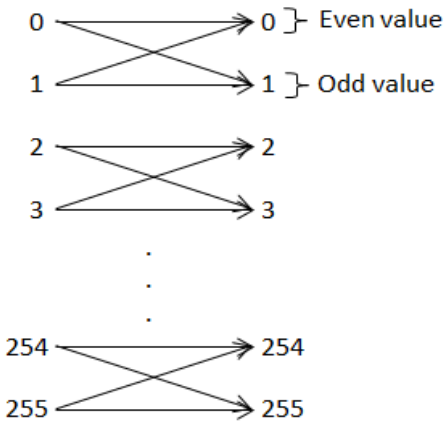


Figure 5.1: Pixel value transitions between cover and stego images with LSB replacement

5.3 Pairs of Values Analysis

Most LSB embedding steganography overwrites the LSB of the pixel values with the secret message bits. This transforms pixel values into other values that differ only in their LSB. These values are known as pairs of values (PoV), as explained in previous section.

The chi-square attack, proposed by (Westfeld & Pfitzmann, 2000), can detect the sequentially embedded LSB steganography in images. The nearly equal distribution of bits in the secret message, especially in encrypted versions, affect the LSB of the pixel values and will generate a close to equal number of occurrences of values in each PoV after embedding. These close to equal occurrences are usually not found in clean images. As the embedding process transforms pixel values into each other in PoVs, the theoretically expected frequency for stego image will be the arithmetic mean of PoVs. Hence, the probability of having secret messages embedded would be measured by the degree of similarity between the theoretically expected frequency and the observed sample distribution, as explained below:

- The method considers K categories (K=128 for 8-bit pixel values) of PoVs and each observed pixel value from the image lies in one of them, for example values from (2k and 2k+1) will fall in category k (where in this case; $0 \leq k \leq 127$).
- The arithmetic mean of occurrences in each PoV represents the theoretically expected frequencies; any values of theoretically expected frequencies less than 5 will be omitted.

$$n_k^* = \frac{|\{color|sortedIndexOf(color) \in \{2k,2k+1\}\}|}{2} \qquad \textbf{(5.1)}$$

- Without loss of generality, the even values of frequency of occurrences in the observed sample have been taken in each PoV and measured by the following:

$$n_k = |\{color|sortedIndexOf(color) = 2k\}| \qquad \textbf{(5.2)}$$

- Then the chi-square (X^2) is applied with $k-1$ degree of freedom:

$$X_{k-1}^2 = \sum_{i=1}^{K} \frac{(n_i - n_i^*)^2}{n_i^*} \qquad \textbf{(5.3)}$$

- The integration of the density function is used to find the probability of embedding (P), assuming the equal distributions of n_i and n_i^*:

$$P = 1 - \frac{1}{2^{\frac{k-1}{2}}\Gamma\left(\frac{k-1}{2}\right)} \int_0^{X_{k-1}^2} e^{-\frac{x}{2}} x^{\frac{k-1}{2}-1} dx \qquad \textbf{(5.4)}$$

The probability of embedding P becomes nearly 0 when X_{k-1}^2 approaches to infinity, and it approaches to 1 for small value of X_{k-1}^2.

5.4 Extended Pairs of Values

Embedding in 2LSB can be divided into two major types (Andrew D Ker, 2007c; K. Zhang et al., 2009): 2LSB and I2LSB. 2LSB directly replaces both 2LSB of the selected pixel values with two bits of the secret message, while I2LSB replaces the 2LSB of the chosen pixel values independently. For example, it can start with replacing the first LSBs of all selected pixel values and then the second LSBs separately, or vice versa.

Both methods of 2LSB steganography, 2LSB and I2LSB, are clearly transferring pixel values into each other in such a way that their pixel values are different only in their first and/or second LSBs. This transition bounds the pixel values into groups of four, as shown in Figure 5.2. It also leads to breaking the correlation between 7th and 8th bit-planes in each pixel value, by inserting random stream of binary values. However, this correlation between 7th and 8th bit-planes is not random in clean images, which would be the base for the proposed detection method.

The 2LSB steganography transfers the pixel values into each other in such a way that they will be different only in their 2LSB values. Hence, it bounds the transitions into groups of four values (4k, 4k+1, 4k+2, and 4k+3), where k value ranges from 0 to 63, as shown in Figure 5.2. These groups of four values will be named as Extended Pairs of Values (EPoV), which could be used as a base for proposing a new 2LSB detection method.

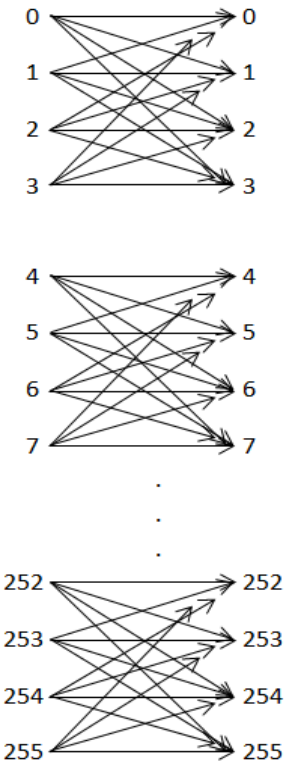


Figure 5.2: Pixel value transitions between cover and stego images with 2LSB replacement

The sum of frequency of occurrences of pixel values within each EPoV will stay the same for a specific image before and after 2LSB embedding has taken place. Also, data embedding in any bit-plane lowers down its correlation with other contiguous bit-planes (Avcibaş et al., 2005), and the data to be hidden is usually more random than the cover component it replaces (Wayner, 2002). Hence, the correlation between the low bit-planes in a stego image pixel values would be different from its clean version. Basically, as the 2LSB replacement inserts noise signal to the two lower bit-planes, the stego images are expected to have more random binary values in their 2LSB of the pixel values. Thus some statistical analysis of the random 2LSBs of the pixel values could be used to detect the 2LSB embedding.

Starting with some basic investigation of the pixel values within the EPoV would be very useful, which is the comparison between pixel values with the same 2LSB (xxxxxx00, and xxxxxx11) and different 2LSB (xxxxxx01, and xxxxxx10) within each EPoV. Hence, the regularity rate will be used to refer to the average of all rates of same to different 2LSB pixel values in each EPoV, as shown below:

$$\text{Same2LSB}(k) = \text{frequencyOfOccurrence}(4k \,\&\, 4k+3) \qquad \textbf{(5.5)}$$

$$\text{Different2LSB}(k) = \text{frequencyOfOccurrence}(4k+1 \,\&\, 4k+2) \qquad \textbf{(5.6)}$$

For each k, if Same2LSB(k) or Different2LSB(K) values were 0, then it set them both to 1 to eliminate the effect of this type of occurrence and improve the accuracy. Now, the regularity rate would be the average of all rates between same to different 2LSBs in each category, as shown in the equation below, where K=64:

$$\text{RegularityRate} = \frac{\sum_{k=0}^{(K-1)} \text{Same2LSB(k)/Different2LSB(k)}}{K} \qquad \textbf{(5.7)}$$

Naturally more random 2LSBs are expected in the stego image pixel values than in the clean version. In other words, more rates of different 2LSBs (xxxxxx01, and xxxxxx10) are expected than the same 2LSBs (xxxxxx00, and xxxxxx11) in stego image pixel values. For this purpose, 49303 images are analysed; 24,761 of them were random images from Google (O. S. Khalind, Hernandez-Castro, & Aziz, 2013), 19,392 images were from ASIRRA (Animal Species Image Recognition for Restricting Access) public corpus pet images (Douceur et al.), and 5,150 never-compressed images from Multimedia Forensics Group image database of Sam Houston State University ("Never-compressed image database,").

On average, 96.8% of them had a higher rate of same 2LSB (xxxxxx00, xxxxxx11) than different ones (xxxxxx01, xxxxxx10) in their pixel values (RGB) for each EPoV, as shown in Table 5.1. However, almost all the regularity rates have a very close values to 1, which means that the rate of same to different 2LSBs pixel values are very close, considering the EPoV. These rates completely change after 2LSB steganography has taken place, as shown in section 5.6.2.

The sequential 2LSB embedding is considered, starting from the top-left pixel to the bottom-right, and each 2LSB of the pixel value is replaced with 2-bits of the random message.

Table 5.1: The percentage of all clean images with overall regularity rates equal to or greater than 1

Image Group	**No. of Images**	**Red**	**Green**	**Blue**
Random images from Google	19392	98.0%	96.9%	98.5%
ASIRRA pet images	24761	95.6%	93.7%	96.5%
Never-compressed images	5150	97.3%	97.2%	97.8%

According to the experimental results, the regularity rate is reduced after converting them into grey-scale. As two sets of 3000 images are considered for experimental results, compressed and uncompressed, to evaluate the proposed method, both sets are converted into grey-scale images and Table 5.2 shows the regularity rate of the grey-scale version. This reduction results from the

calculation of transforming the pixel values from RGB to grey-scale. However, the regularity rates were again very close to 1.

Table 5.2: The percentage of all clean images with overall regularity rates equal to or greater than 1

Image Group	No. of Images	Regularity Rate
ASIRRA pet images	3000	82.4%
Never-compressed images	3000	79.4%

5.5 Steganalysis of 2LSB Embedding Method

The steganalysis methods of 2LSB detection are quite new, and during the past decade a number of detection methods have been proposed to detect extended methods of LSB. Some methods proposed to detect multiple LSB steganography which are expected to have lower accuracy than the 2LSB specific steganalysis methods. Other methods are specific to the detection of 2LSB steganography, as explained below.

(Luo et al., 2006) proposed a detection method of 2LSB steganography in digital images based on quartic equation. The method constructs a finite state machine based on the sample pairs of the image pixel values, and then builds a quartic equation via the relation of the conversion states to obtain the estimated embedding rate. However, as claimed by Niu et al. (Niu et al., 2009), the calculations are too complex and take too long time for analysis purposes. Another drawback is that the authors took a set of only 100 images for testing without showing the ROC graph to show the performance of the classifier.

Ker (Andrew D Ker, 2007c) also proposed a steganalysis method to detect 2LSB message embedding in digital images by extending the structural analysis of the image. This method uses statistics of many variances to form the equation that estimates the message length. This method is also considered as a complex detection method because it involves lots of calculations. This method has been superseded (i.e. outperformed) by the detection method proposed by Niu et al. (Niu et al., 2009), discussed below.

Another method of detecting 2LSB steganography was proposed by Zhang et al. (K. Zhang et al., 2009) based on the statistical characteristics in the 2LSB of the pixel values in the image. The detection accuracy can reach 90% only when the embedding rate is 0.2 or more. Thus it limits the performance of detection for lower embedding rates.

The last and the most accurate detection method of 2LSB steganography in digital images was proposed by Niu et al. (Niu et al., 2009). They estimate an approximately the cover image through a local masked estimation function, and then they construct a weighted stego image. The equation of detection is formulated as a simple optimisation problem between weighted stego and approximately cover image. This method can accurately detect and estimate the length of the embedded message by constructing a weighted stego image and using least square equation. The authors compared their results with the detection method proposed by Ker (Andrew D Ker, 2007c) and demonstrated better accuracy and faster detection for the same set of images. Hence, this method is considered to be compared with the accuracy of the proposed method.

Also, there are some methods for detecting multiple LSB embedding steganography of which 2LSB is a component, including WS (Fridrich & Goljan, 2004), which was extended by Yu et al. (X. Yu et al., 2005) to detect n-LSB steganography, which could also estimate the message length. This method, as claimed by Yu et al. (Xiaoyi Yu & Babaguchi, 2008), has drawbacks of low accuracy and assumptions like a symmetric property in the pixels of the cover image.

Another estimation method of detecting n-LSB embedding was proposed based on WS image (Xiaoyi Yu & Babaguchi, 2008), which puts no assumption on the cover image. As claimed by the authors, their method has very low computation complexity with a clear estimation formula. The method could accurately detect the existence of the secret message and estimate the embedding ratio.

Also, a method of detecting MLSB (multiple least significant bits) steganography was proposed by Yang et al. (Yang et al., 2008) based on the transition relationships among some trace subsets. The method could estimate the amount of embedded secret messages and is also defined as a very accurate method of detection by the author.

Based on SP analysis, Luo et al. (Luo et al., 2012) proposed a method to estimate the embedding ratios of multiple bit-planes image steganography combining suitable trace sets to estimate the modification ratios in grey code bit-planes. As claimed by the author, the proposed method can estimate the embedding ratios of multiple bit-planes with smaller errors in comparison to previous steganalysis methods.

We propose the concept of extended pairs of values (EPoV) that relies on the arithmetic mean of the histogram of each group of extended pairs, which stays unmodified after embedding process. So, it is expected to be more accurate especially for low embedding rates.

5.6 EPoV Analysis and the Chi-square

Changes to two LSB values are much harder than one, due to complex changes in pixel values. Like conventional PoV analysis, this method uses the chi-square attack with a new form of EPoV. Each EPoV consists of four values based on the fact that 2LSB steganography changes these four values into each other, as shown in Figure 5.3. It can be seen that the sum of frequency of occurrences in each EPoV remains constant, before and after embedding, as it puts boundaries for each group and values are changing within these scopes. Moreover, because the embedding process inserts noise to the pixel values, it is expected to have more frequencies of different 2LSB values (xxxxxx01, xxxxxx10) than the same 2LSB values (xxxxxx00, xxxxxx11) in each EPoV.

The seventh and eighth bit-planes of pixel values are not totally random in clean images, which will be the case after embedding process has taken place. Hence, it is uncommon for observed EPoVs (4k+1, 4k+2) and (4k, 4k+3) to be far from their arithmetic means in clean images. Also, for clean images it is more likely to have a higher rate of similar 2LSB values in each EPoV. Thus, the theoretically expected frequency after embedding would be far from the arithmetic mean of the values in each EPoV. Because of this, the arithmetic mean is still considered important in each EPoV. The similarity measure between the observed sample and the arithmetic mean would be the base of detection; being close to arithmetic mean indicates that the image is clean. Otherwise, if it was far from the arithmetic mean, this indicates the existence of hidden content.

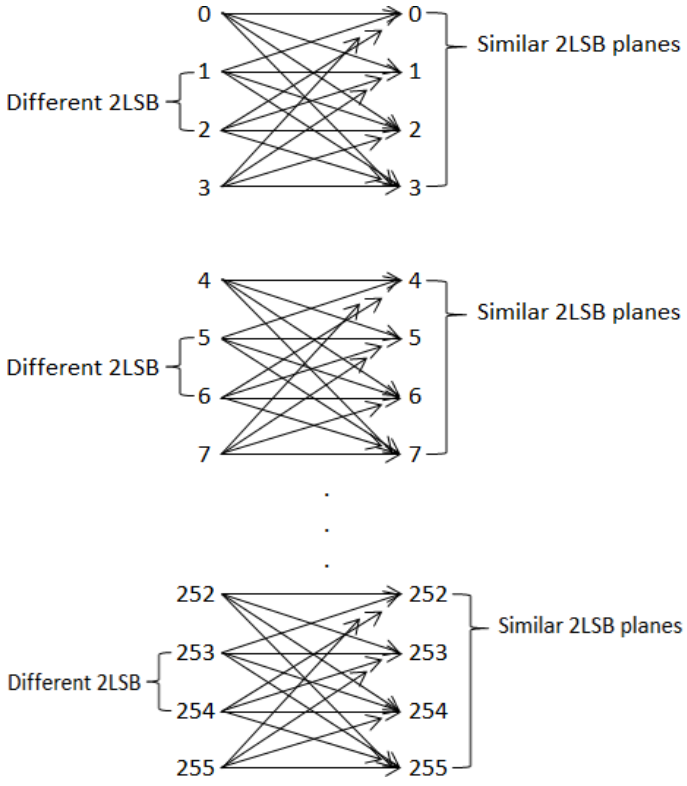


Figure 5.3: Possible transitions and grouping of pixel values with 2LSB embedding

The proposed method of detecting 2LSB steganography uses the chi-squared attack, as explained below.

- As shown in Figure 5.3, K categories of extended PoVs were considered. Since it groups every four values in one category and pixel values (or colour component values RGB) ranging from 0 to 255, the value of K=64. Each colour value from the image pixels lies in one of those EPoVs, such as the values (4k, 4k+1, 4k+2, and 4k+3), all of which belong to category k.
- Two vectors with K elements are used, X^{64x1} and Y^{64x1}, such that:

$$X_k = \text{frequency}(4k \text{ and } 4k+3); 0 \leq k \leq 63$$

$$Y_k = \text{frequency}(4k+1 \text{ and } 4k+2); 0 \leq k \leq 63$$

- The frequency of values with similar 2LSBs in each category is held by X, and different 2LSBs by Y.
- Without loss of generality, this method considers the similar 2LSB values in the EPoVs in such a way that X_k measures the frequency of occurrences in category k.
- The theoretically expected frequency of occurrences for a stego image should be far from the arithmetic mean in each category. However, this is not the case for clean images, which are closer to the arithmetic mean. That is why the arithmetic mean of each category is vital and calculated as follows:

$$Z_k = \frac{X_k + Y_k}{2} \qquad \textbf{(5.8)}$$

- To measure the degree of similarity between the observed frequency of occurrence and the arithmetic means, the chi-squared (X^2) is applied with $k-1$ degree of freedom:

$$X_{k-1}^2 = \sum_{i=1}^{K} \frac{(X_i - Z_i)^2}{Z_i} \qquad \textbf{(5.9)}$$

- Unlike clean images, the X_{k-1}^2 is expected to be relatively high for stego images, as the X_i should be relatively far from Z_i.
- The probability of embedding P is calculated by integration of the density function with an upper limit of X_{k-1}^2, under the condition that the distributions of X_k and Z_k are not equal and relatively highly different.

$$P = \frac{1}{2^{\frac{k-1}{2}}\Gamma\left(\frac{k-1}{2}\right)} \int_0^{X_{k-1}^2} e^{-\frac{u}{2}} u^{\frac{k-1}{2}-1} du \qquad \textbf{(5.10)}$$

The probability of embedding P converges to 1 as X_{k-1}^2 approaches infinity, and for relatively small X_{k-1}^2 becomes much less than 1, which is affected by embedded message.

Hence, the key difference between the proposed method and the conventional PoV analysis is that it creates a new form of pairs of values which reduces the number of categories from 128 to 64. Also, it assumes a noticeable difference between the observed frequency of occurrence and

the arithmetic mean in each category. Finally, it finds the final probability of embedding (P) from the average of all Ps calculated from 1% to 100% of the entire image separately.

To analyse the image, the method checks the value of P from (1% - 100%) of the total image pixels. The continuity of P being equal to 1 within the entire image shows the availability of hidden content, otherwise the image is considered as clean. To visualise this we analysed the standard image of Lenna 512x512 twice, before and after embedding for colour and grey-scale versions (see Figure 5.4 to Figure 5.7). However, according to experiments, the value of P will not be stable until 5% of the images' total pixels are analysed. As a refinement of the results we omitted the first 4% in finding the final value of P, which become the average of all Ps from 5% to 100% of the image's total pixels.

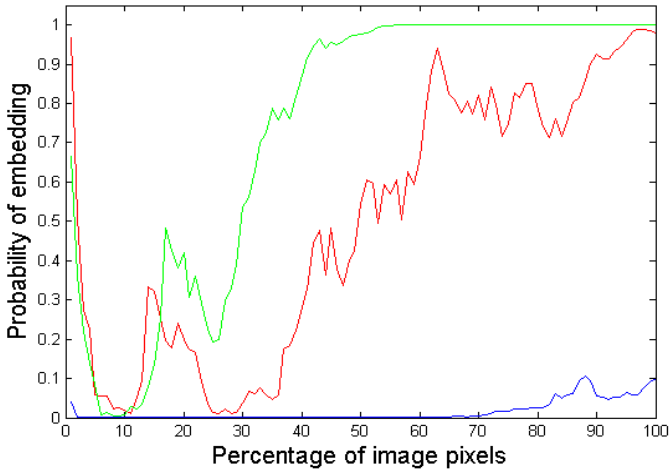


Figure 5.4: The probability of embedding for Lenna's 512x512 colour clean image

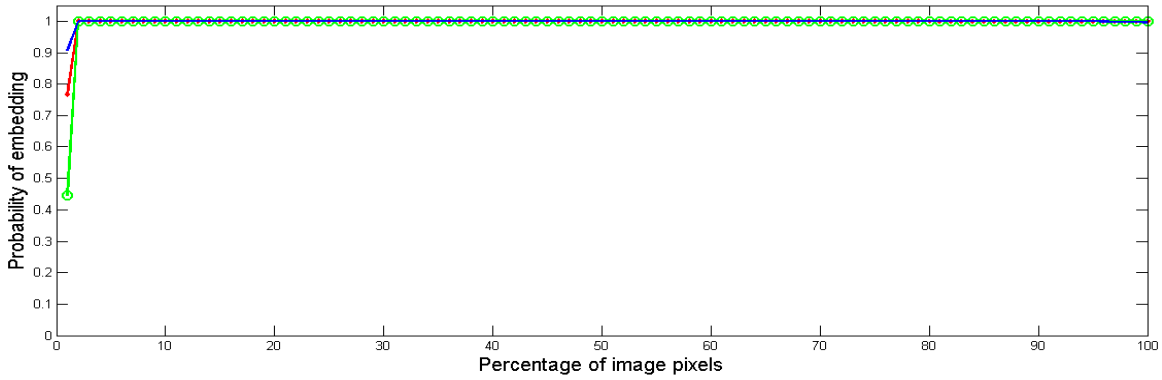


Figure 5.5: The probability of embedding for Lenna's 512x512 colour stego image

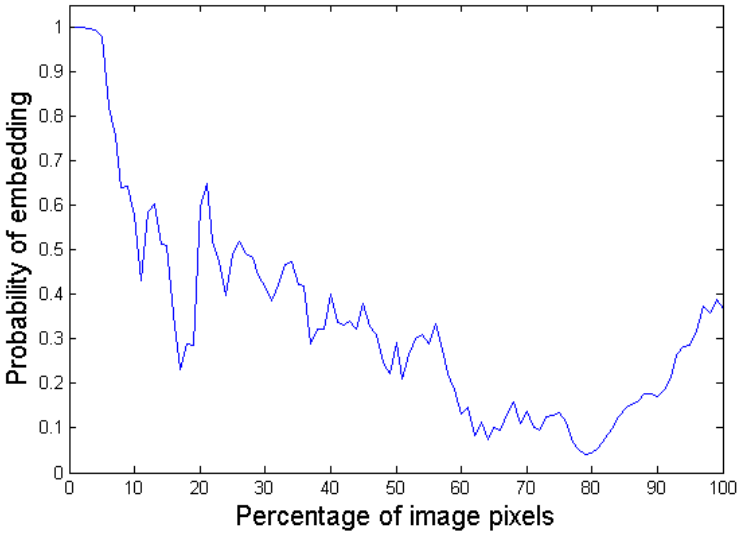


Figure 5.6: The probability of embedding for Lenna's 512x512 grayscale clean image

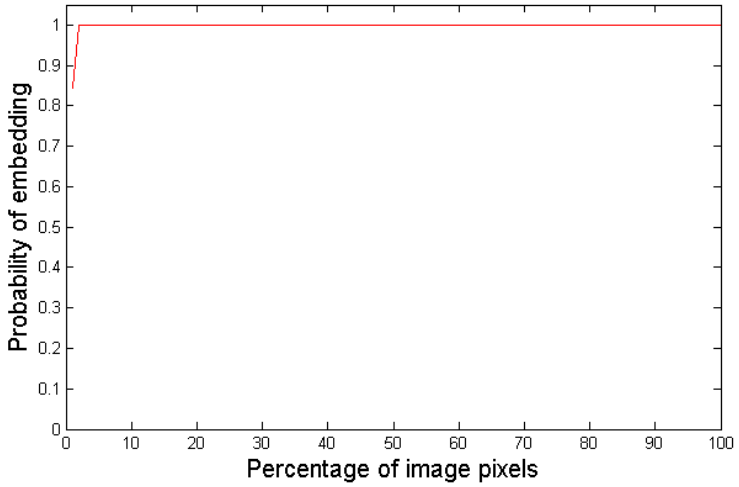


Figure 5.7: The probability of embedding for Lenna's 512x512 grayscale stego image

It can be seen from Figure 5.4 and Figure 5.5 that the P value in the case of the clean image varies in all colour components (RGB), as coloured accordingly, and for the stego image it continues being 1 for all colour components. This is also the case with the grayscale version of the Lenna image, as shown in Figure 5.6 and Figure 5.7.

5.6.1 Experimental Results

To analyse the performance of the proposed 2LSB steganalysis method, two sets of compressed and uncompressed images are considered to show the differences via their experimental results. The first set is a group of 3000 compressed random images from ASIRRA pet images (Douceur et al.), taken as cover objects after converting them into grayscale. The reason of choosing this set of images is that they are random images, originally taken from petfinder.com. Although this will negatively affect the performance of the proposed method, it is practical, especially for digital

forensics analysis. The second set is a group of 3000 never-compressed images ("Never-compressed image database,") from Multimedia Forensics Group image database of Sam Houston State University, as they are considered as normal images.

Each image is loaded with a random message with a certain percentage of the total capacity (5%, 10%, 20%, 30%... 100%). For each percentage the detection method is fed with 6000 images; 3000 stego images with the specified amount of embedded message together with 3000 original ones for classification. Each image set is examined separately, as follows.

5.6.1.1 Compressed Images

The performance of the detection method is evaluated in two perspectives, as a discrete classifier, as in Table 5.3, and as a continuous classifier, like in Figure 5.8. According to the experimental results shown in Table 5.3, the detection method can accurately detect 2LSB replacement, especially when the embedding rate reaches 10% of the image's total capacity. The true positive rate was very high, especially for the message length of 20% to 100%, which was 0.997 to 0.999 with accuracy greater than 0.96. Also, the false positive rate was 0.074, which is very low in comparison to the very well-known steganalysis tools like Stegdetect, which scored 0.1 for a random set of images from Google for the default sensitivity value of 1 (O. S. Khalind et al., 2013).

Table 5.3: The experimental results of compressed images; alerts, positive rates, and accuracy

Embedded data amount	True +	True -	False +	False -	True + Rate	False + Rate	Accuracy
5%	815	2778	222	2185	0.272	0.074	0.599
10%	2499	2778	222	501	0.833	0.074	0.879
20%	2991	2778	222	9	0.997	0.074	0.962
30%	2996	2778	222	4	0.999	0.074	0.962
40%	2995	2778	222	5	0.998	0.074	0.962
50%	2998	2778	222	2	0.999	0.074	0.963
60%	2996	2778	222	4	0.999	0.074	0.962
70%	2998	2778	222	2	0.999	0.074	0.963
80%	2997	2778	222	3	0.999	0.074	0.963
90%	2995	2778	222	5	0.998	0.074	0.962
100%	2997	2778	222	3	0.999	0.074	0.963

There are some very small variances in number of true positives between different amounts of embedded data, especially from 30% to 100%, which result from the randomness of embedded messages for each case. Also, as it could be noted, the False positive rate is constant because for every embedding rate we have the same set of clean images.

The performance of the classifier is shown in Figure 5.8, in the form of ROC curve. The straight line from (0, 0) to (1, 1) indicates the random guess. Any curve located above this line is considered as better than random guess, and a larger the area under the curve indicates better performance of the classifier. For the proposed method, there are three curves labelled with the specified percentages (5%, 10% and 20-100%), so the classifier was in its best performance when the amount of data was from 20% to 100%.

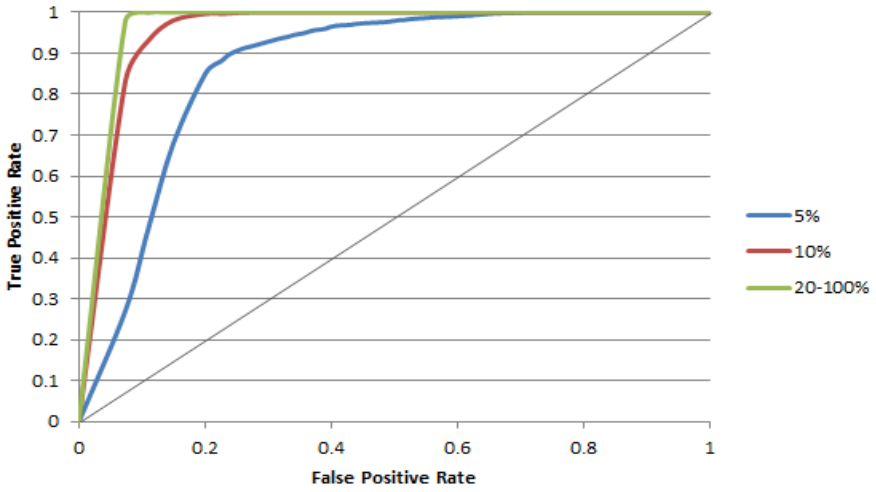


Figure 5.8: The ROC curve of the compressed image set

5.6.1.2 Never-Compressed Images

Again, both discrete and continuous classifier types are considered. The experimental results for uncompressed image set were slightly different from the compressed one. The experimental results showed that the detection threshold value should be reduced from 1 to 0.99 to get the highest accuracy due to having a weaker (lower) bit-planes similarity in never-compressed images, as their pixel values are taken without any sort of image processing.

Again, the proposed method can accurately detect the existence of the secret message, especially from the embedding rate of 10%. For the embedding rates of 20% - 100%, the true positive rate was very high at 0.999 to 1, with accuracy of more than 0.98. As shown in Table 5.4, the false positive rate were only 0.036, which is even less than the compressed set. Again, the false positive rate is constant because for every embedding rate we have the same set of clean images.

Table 5.4: The experimental results of uncompressed images; alerts, positive rates, and accuracy

Embedded data amount	True +	True -	False +	False -	True + Rate	False + Rate	Accuracy
5%	561	2890	110	2439	0.187	0.036	0.575
10%	2249	2890	110	751	0.749	0.036	0.857
20%	2998	2890	110	2	0.999	0.036	0.981
30%	3000	2890	110	0	1	0.036	0.982
40%	3000	2890	110	0	1	0.036	0.982
50%	3000	2890	110	0	1	0.036	0.982
60%	3000	2890	110	0	1	0.036	0.982
70%	3000	2890	110	0	1	0.036	0.982
80%	3000	2890	110	0	1	0.036	0.982
90%	3000	2890	110	0	1	0.036	0.982
100%	3000	2890	110	0	1	0.036	0.982

As could be noticed from Figure 5.9, the proposed method was more accurate on the never-compressed image set, except for the embedding rate of 5%, which was better in the compressed image set.

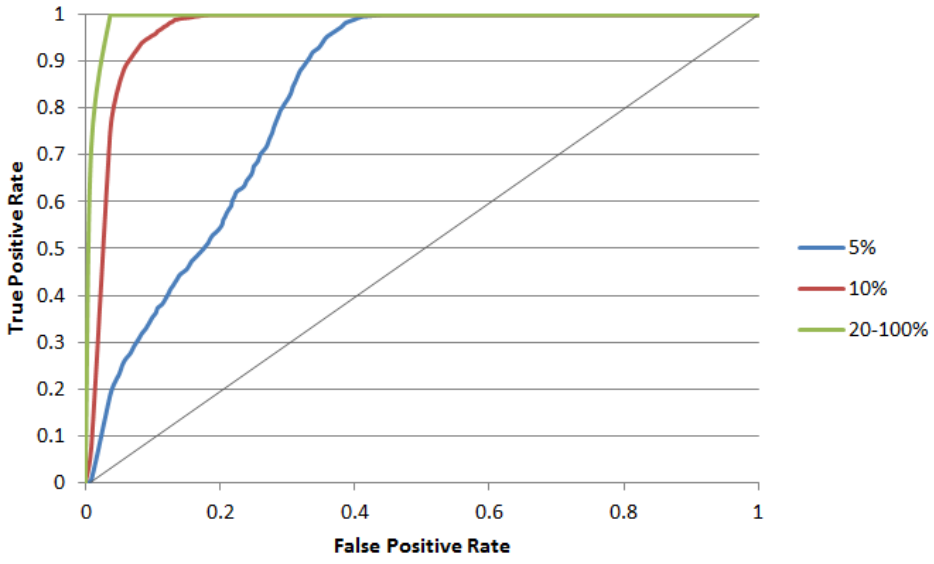


Figure 5.9: The ROC curve of the uncompressed image set

5.6.2 Estimating the Message Length

The detection method only gives a decision of 'Stego' or 'Clean' for each image without specifying the embedded message length. This helps the proposed method to be used as an automated tool for analysing a bulk of images and reduce an overhead of setting the appropriate threshold by the digital forensics analyst.

To estimate the embedded message length, the regularity rate is considered, which is directly affected by the amount of embedded message, as in equation 5.7. Based on the experimental results of our set of 6000 images, as shown in Table 5.5 and Table 5.6, the regularity rate can be divided into five ranges: less than 0.65, 0.65-0.8, 0.8-0.95, 0.95-1, and greater than 1.

Table 5.5: Regularity rate versus embedding rate for compressed image set

Message amount	**Regularity Rate**				
	< 0.65	0.65-0.8	0.8-0.95	0.95-1	> 1
0%	0.00	0.00	0.00	0.18	0.82
5%	0.00	0.00	0.05	0.52	0.43
10%	0.00	0.00	0.31	0.45	0.23
20%	0.00	0.01	0.74	0.19	0.07
30%	0.00	0.10	0.83	0.05	0.02
40%	0.00	0.45	0.53	0.01	0.01
50%	0.01	0.78	0.21	0.00	0.00
60%	0.11	0.82	0.06	0.00	0.00
70%	0.46	0.52	0.02	0.00	0.00
80%	0.77	0.23	0.00	0.00	0.00
90%	0.91	0.09	0.00	0.00	0.00
100%	0.97	0.03	0.00	0.00	0.00

Table 5.6: Regularity rate versus embedding rate for uncompressed image set

Message amount	**Regularity Rate**				
	< 0.65	0.65-0.8	0.8-0.95	0.95-1	> 1
0%	0.00	0.00	0.00	0.21	0.79
5%	0.00	0.00	0.09	0.52	0.39
10%	0.00	0.00	0.42	0.38	0.20
20%	0.00	0.02	0.78	0.13	0.07
30%	0.00	0.15	0.76	0.05	0.03
40%	0.01	0.52	0.43	0.02	0.02
50%	0.03	0.75	0.21	0.01	0.01
60%	0.12	0.76	0.10	0.01	0.01
70%	0.45	0.49	0.05	0.01	0.00
80%	0.72	0.25	0.03	0.00	0.00
90%	0.84	0.14	0.02	0.00	0.00
100%	0.91	0.07	0.01	0.00	0.00

Each value of regularity rate represents the percentage of values within the specified range. Based on Table 5.5 and Table 5.6, we can derive another table that maps the regularity rate with the embedding rate (Table 5.7). The proposed method can now accurately estimate the embedded message length. Of course, there is some overlap between certain ranges of the regularity rate and the message size, but the boundaries could still be identified with a high level of certainty.

Table 5.7: Regularity rate and the amount of embedded message

Regularity Rate	**Estimated amount of embedded data**
larger than 1	0%
between (0.95 – 1)	5% - 10%
between (0.8 – 0.95)	20% - 40%
between (0.65 – 0.8)	50% - 70%
Less than 0.65	80% - 100%

5.7 EPoV Analysis and the Standard Deviation

The 2LSB steganography results in more complicated changes on the intensity histogram of the pixel values than LSB methods, which makes the detection process harder to perform. It changes the two lower bit-planes (7^{th} and 8^{th}, for 8-bit pixel values) and bounds the transition of pixel values into groups of four values called EPoV, which can be used for detection analysis (O. Khalind & Aziz, 2014). Instead of separating the detection from the estimation of the hidden message length, here another method is proposed to measure the amount of change in the stego image by the embedding process.

As mentioned earlier, the 2LSB embedding causes the insertion of random sequence of binary values, resulting in a broken correlation in lower bit-planes (7^{th} and 8^{th}), which is not random in clean images. Moreover, it is expected that there will be more different pairs of bit values in lower two bit-planes (xxxxxx01, xxxxxx10) after embedding with 2LSB steganography. Hence, they are grouped into similar (xxxxxx00, xxxxxx11) and different (xxxxxx01, xxxxxx10) 2LSBs pixel values, as shown in Figure 5.3.

If k indicates the index of the EPoVs, which could range from 0 to 63 for 8-bit pixel values, then the same and different 2LSB pixel values within a certain EPoV would be $(4\mathrm{k}, 4\mathrm{k}+3)$ and $(4\mathrm{k}+1, 4\mathrm{k}+2)$ respectively.

The sum of frequency of occurrences within each EPoV stays unchanged before and after the embedding process. Hence, taking the arithmetic mean of the frequency of occurrences of both same and different 2LSB pixel value groups in each EPoV would be considered to measure the imbalance between same (00, 11) and different (01, 10) 2LSB in the image before and after the embedding place has taken place.

According to analysing 3000 never-compressed images, more than 97% of the standard deviation of the set of arithmetic means of frequency occurrences for both groups (same and different) in each EPoVs was very close to the standard deviation of the set of frequency of occurrences for the same 2LSBs in each EPoVs. Thus, dividing the standard deviation of the arithmetic means of both groups in the EPoVs by the standard deviation of the same 2LSBs group would be very close to 1 in clean images.

Based on this conclusion, to find the amount of changes by the 2LSB embedding, we subtract the expected value of the clean image, which is 1, and the remaining will be the modification rate. According to the experimental results this value will reach up to 1.5 in the corresponding stego image for the embedding rate of 1. So, the expected value for clean images will be subtracted from the observed value and the result will be 0.5, which implies that half of the image is modified. In other words, the total capacity of the image has been used by the embedding process. Hence, the modification rate, after subtracting 1, ranges from 0 to 0.5, which is directly proportional to the embedding rate. The detection process is shown in Figure 5.10.

Input: Image I
Output: Double modificationRate
Start
Array $X_{64} = 0, Y_{64} = 0$
For all pixel values P of I
    $indexOfEPoV = P/4 + 1$
    If $2\text{LSB}(P) = 11\ OR\ 2LSB(P) = 00$
        Increment X(indexOfEPoV)
    else
        Increment Y(indexOfEPoV)
    End
    For i=1 to 64
        $Z = round\left(\left(X(i) + Y(i)\right)/2\right)$
    End
End
$modificationRate = std(Z)/std(X) - 1$
End

Figure 5.10: The pseudo-code of detection algorithm

Figure 5.11 and Figure 5.12 clearly show the differences between the clean and stego versions of the Lenna image. They show the frequency of occurrences for the same 2LSB pixel values (X), the arithmetic mean of same and different 2LSB pixel values (Z) in each EPoVs, and their standard deviation. It can be seen that they are very close for the clean version of the image and different for the stego version, with an embedding rate of 1. Moreover, as shown in Figure 5.13, the detection result is very close to zero for the clean version and 0.5 for the stego version of the Lenna image with the embedding rate of 1, after subtracting the expected value of the clean image, which is 1.

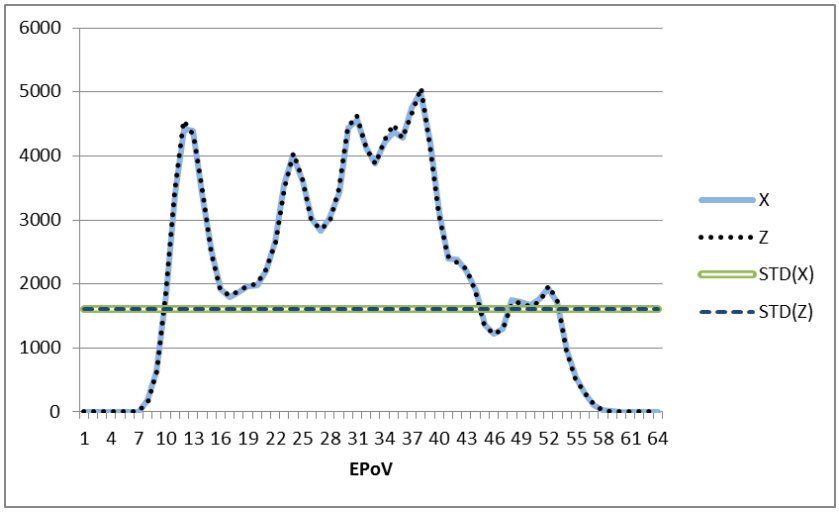


Figure 5.11: Analysis of Lenna clean image

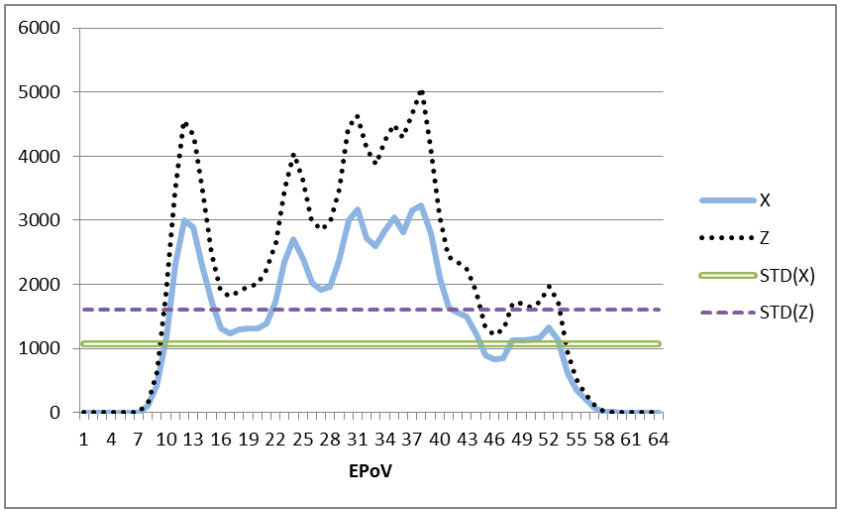


Figure 5.12: Analysis of Lenna stego image with an embedding rate of 1

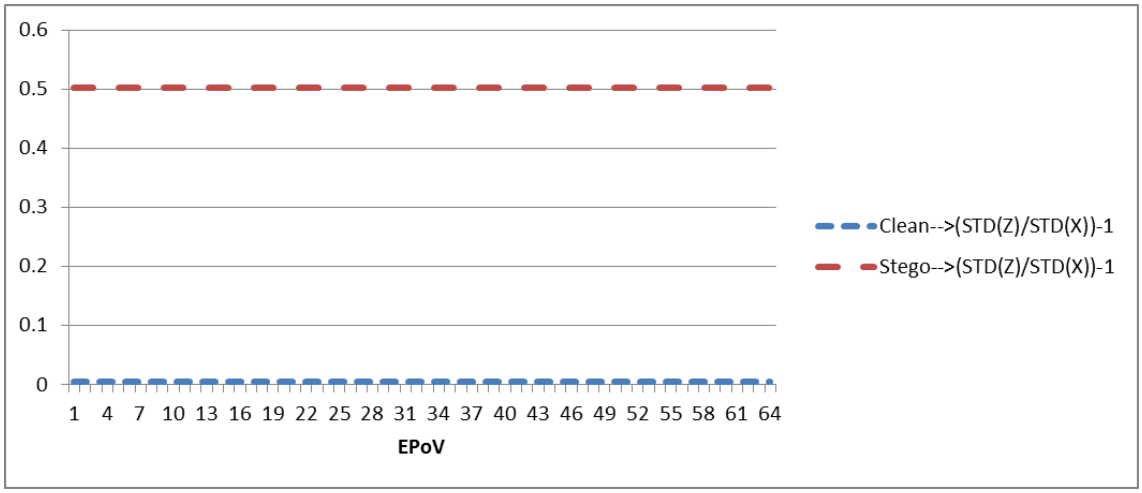


Figure 5.13: The detection results of the clean and stego version of Lenna image

5.7.1 Experimental Results

As a basic evaluation, the three common images among steganographers (Lenna, Pepper and Baboon) are taken into consideration. The results of the estimated amount of the image that has changed are shown in Table 5.8 and Table 5.9 for both proposed method and the existing one (WS2) (Niu et al., 2009).

Table 5.8: Detection results of the proposed method

Images	**Embedding rate**						
	0%	5%	10%	20%	50%	75%	100%
Lenna	0.005	0.031	0.054	0.096	0.205	0.340	0.497
Pepper	0.000	0.020	0.045	0.103	0.246	0.367	0.505
Baboon	0.002	0.015	0.023	0.047	0.168	0.301	0.498

Table 5.9: Detection results of the WS2

Images	**Embedding rate**						
	0%	5%	10%	20%	50%	75%	100%
Lenna	0.008	0.024	0.038	0.072	0.174	0.270	0.385
Pepper	0.007	0.028	0.039	0.076	0.181	0.287	0.398
Baboon	0.028	0.035	0.056	0.086	0.189	0.278	0.403

The estimation of the message length (or the modification rate of the image) could also be evaluated by comparing it with a perfect classifier, which is practically does not exist. Table 5.10 shows the average of differences for all embedding rates of the three images with the perfect classifier. It can be seen that the proposed method is more accurate than the WS2.

Table 5.10: The difference between the detection methods and the perfect classifier

Detection methods	Average difference
Proposed method	0.018
WS2	0.046

To evaluate the proposed steganalysis method, a set of 3000 never-compressed images ("Never-compressed image database,") are used as cover objects after converting them into grey-scale. For each embedding rate (5%, 10%, 20%, 50%, 100%) the images are loaded with a stream of pseudo-random bits as a secret message, to have all the statistical properties of the encrypted version of it (Westfeld & Pfitzmann, 2000). The stego images are then fed into both the proposed method and the most accurate detection method of the targeted 2LSB steganalysis (Niu et al., 2009) for comparison. The results are shown in the form of ROC graphs for both detection methods in Figure 5.14 and Figure 5.15. It can be seen that the proposed method outperforms WS2 for low embedding rates (less than 50%). This is because the weighted stego method relies on the probabilistic model of the cover image, which is expected to not always be very accurate, especially for low embedding rates. However, the proposed method relies on the arithmetic mean of the frequency of occurrences in each EPoV which has the same value for both clean and stego versions of the image with any embedding rate.

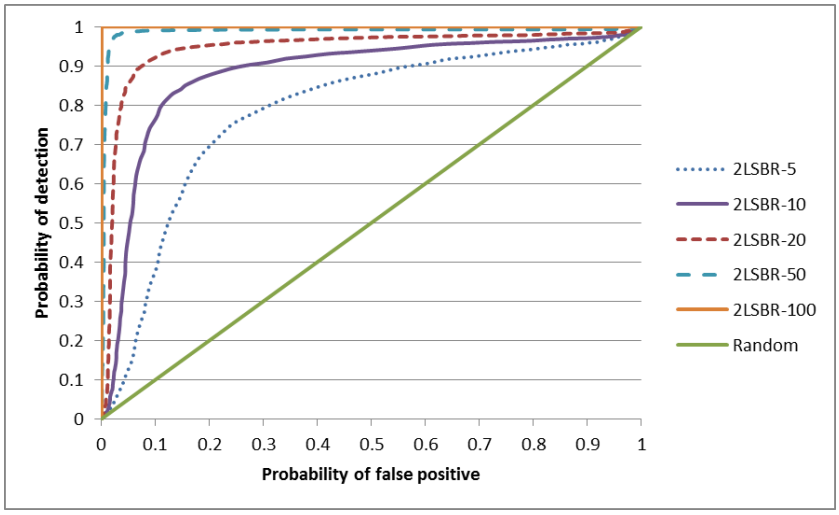


Figure 5.14: The ROC graph of the proposed method for 3000 images

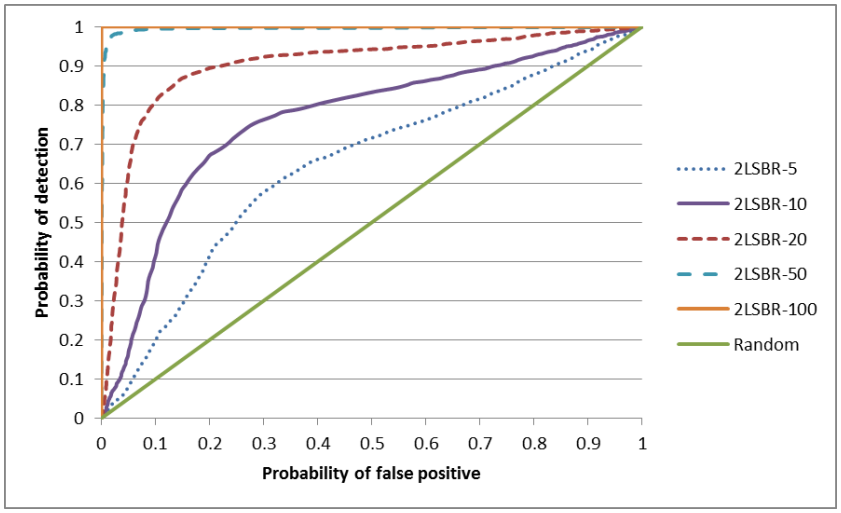


Figure 5.15: The ROC graph of the WS2 for 3000 images

5.8 Conclusion

In this chapter, a new method of detecting 2LSB steganography in still images was proposed based on a new form of pixel value analysis. The EPoV analysis was used twice, with the chi-square attack and the standard deviation.

The chi-square method separates the estimation of the message length from the detection process, which could be used as a discrete classifier by giving a label (Stego or Clean) to the analysed image. This classifier type does not need the setting of the threshold value, which practically becomes more useful for the digital forensics analyst by eliminating the overhead of putting a right threshold value. Moreover, since the practical side is considered, as a discrete classifier, it is tested against two sets of images; the random compressed and uncompressed images. The experimental results showed that the detection method can accurately detect the existence of the secret message, especially when the embedding rate reaches 10% of the image's total capacity. It also could estimate the amount of embedded message in stego images as a second level analysis based on five ranges of regularity rate.

Moreover, the method is very simple to understand and implement without any computational complexity, and it could actively work on both sets of 3000 images, compressed and uncompressed, with random messages. As mentioned earlier, the method could also be applied on colour images to indicate which colour components (R, G and/or B) have been used by the embedding process. Also, it could be used as an automated tool by digital forensics analysts in their investigation process to analyse a bulk of images for hidden contents without the overhead of choosing an appropriate threshold.

The standard deviation method of the EPoV can also accurately detect the 2LSB steganography in digital images. This method gives better accuracy in detection for low embedding rates than existing methods. It can also accurately estimate the length of the hidden message for any embedding rate. Therefore, it can be more useful than the chi-square attack and improve the current accuracy of the 2LSB steganalysis methods in the literature. Moreover, as it considers the arithmetic mean of the frequency of occurrences in each EPoV, which would be the same before and after embedding for a certain image, the proposed detection method can maintain its high accuracy for low embedding rates as well. Also, this method is more consistent with other detection methods in the literature by giving the image membership probability to the stego class. Thus, it acts as a continuous targeted 2LSB steganalysis method.

CHAPTER 6: THE FORENSIC EVALUATION OF STEGANALYSIS TOOLS

6.1 Introduction

The evaluation of any steganalysis tool is a difficult task and also a very time-consuming process. In chapter three, the performance evaluation of steganalysis methods was shown for very simple detection methods. However, the traditional evaluation process of steganalysis tools could be more complicated when the steganalysis tools are capable of detecting multiple embedding techniques. One such tool is Stegdetect (N Provos, 2008).

Preparing a suitable testing set for image steganalysis tools is a challenging task, as it needs the choice of embedding process and embedding rate. This could be even more challenging if the evaluation of the steganalysis tool is done by digital investigators, who are expected to use steganalysis tools as a black box without knowing the very technical details of the detection method.

Another reason is that most authors of steganography and steganalysis demonstrate their techniques under laboratory conditions, thus practical applications remain largely unsolved (Andrew D Ker et al., 2013) since the embedding algorithm, the source of cover objects, and the objects to be examined are perfectly known in laboratory simulations. Thus normal images in steganography laboratories (uncompressed, non-modified, and without any processing on the image) do not reflect normal images in real-world. For example, the digital forensic investigator may analyse a bulk of digital files on a suspicious storage media. One might expect to find these public images, audio files, games, applications and documents (which could be compressed or modified). Hence, from the digital forensics point of view, it could be possible to say that the random look of the files could prove the nonexistence of the hidden communication.

If the model of covers is absent, the feature-based steganalysis and machine learning could act as the best detection method of image steganography (Andrew D Ker et al., 2013). It represents the media using a much smaller dimensionality feature, then after creating a training database from cover and stego examples, a binary classifier is trained to differentiate these two classes. However, the correct cover source highly affects the detection accuracy. Hence, the reliability of detection varies from a training database to another. Therefore, the classifier may suffer from decreased accuracy because of the cover source mismatch.

Moreover, it could be even more challenging for digital forensic investigators, because everyone has a certain set of digital files (images, audio files, games etc.) on their devices, which might be

downloaded from different sources over the Internet. All these digital media files are certainly not clean, they may contain hidden contents for copyright protection or ownership fingerprints etc. Thus it is highly likely to have hidden contents in some media files without any connection to cybercrimes, hence it is better to consider the similarity with a random set of digital media as a baseline, and not only a pure laboratory-based set of digital media.

As a result, real-world steganography can be expected to be difficult to detect. Hence, real-world steganalysis is required to have very low false positive rates (Andrew D Ker et al., 2013). Hence, this chapter considers the study of false positive rate of Stegdetect; a well-known image steganalysis tool. In the study, more than 40,000 images were processed, which were randomly downloaded from the Internet using Google Images, together with 25,000 images from ASIRRA, as explained previously. The aim of this study is to help digital forensic analysts, aiming to study a large number of image files during an investigation, to better understand the capabilities and the limitations of steganalysis tools like Stegdetect.

The results obtained show that the rate of false positives generated by Stegdetect depends highly on the chosen sensitivity value, and it is generally quite high. This should support the forensic expert to have better interpretation in their results, and to consider the false positive rates during their investigations. Additionally, a detailed statistical analysis is provided for the obtained results to study the difference in detection between selected groups (close groups and different groups of images). This method can be applied to any steganalysis tool, which gives the analyst a better understanding of the detection results, especially when they have no prior information about the false positive rate of the tool.

6.2 Steganalysis Tool Assessment

Steganalysis methods could be modelled as a classification problem, and the output of analysing a bulk of digital media files could be a true positive, false positive, true negative, or false negative. These terms are explained in chapter three with regard to the confusion matrix (Fawcett, 2003).

A lot of performance measures can be calculated from the confusion matrix, including precision, accuracy, F1 score and error rate. Unfortunately, assuming the predictive accuracy as the best way to measure the performance of the classifier is not necessarily true, as it highly depends on the comparative size of the actual stego and clean instances (Max, 2007).

The ROC graph is also another way of showing the performance of the steganalysis classifier via a two-dimensional graph between the false positive and true positive rates on the horizontal and

vertical accesses, respectively (Fawcett, 2003). The classifier with a larger area under the plotted curve represents better detection accuracy. As mentioned earlier, all these evaluation methods are performed in laboratory settings, which do not reflect the real-world evaluation of steganalysis tools.

We use Stegdetect, instead of EPoV steganalysis method proposed in chapter five, as a case study for the proposed statistical analysis of the detection results. This is because the Stegdetect has a wider range of detecting steganographic techniques, in which we can get a larger set of detection results to be fed to the proposed statistical method. Also, since it can analyse the lossy compressed images like JPG, it would make the investigation process be closer to the real-world and not work under the in-laboratory condition.

6.3 Stegdetect

A number of steganalysis tools (software) are available on the web for different types of embedding algorithms and for various digital media. This research focuses on Stegdetect, an automated tool developed to detect hidden content in digital images. Stegdetect can detect secret content in images embedded with a number of different steganographic tools like jsteg, jphide, outguess, f5, appendX, camouflage and alpha-channel (N.-I. Wu & Hwang, 2007). Moreover, it also shows the level of confidence in its detection by appending stars (*), (**), (***). A single star means low confidence and three stars mean high confidence.

Stegdetect uses statistical test for detecting hidden contents and is capable of finding the method used in the embedding process. It is a very popular tool among security and forensic practitioners and can be considered a de facto standard due to its excellent capabilities and the fact that it is a free and open source. There are some options that could be set during the testing phase. This study focuses on the sensitivity option, as it greatly affects the sensitivity of the detection algorithm. The default sensitivity value is 1.0, as highlighted in Table 6.2 and Table 6.6; we explore the whole range (0.1 – 10.0) permitted by Xsteg- the graphical user interface (GUI) of Stegdetect. As claimed by (Cole & Krutz, 2003, p. 209), the value of the sensitivity parameter should be set carefully as it affects both the false positive and false negative rates.

Stegdetect outputs the list of all steganographic methods found in each image which could be negative, appended alpha-channel, camouflage, false positive or others like jphide, outguess, jsteg, and f5, with the confidence level shown by appended stars. (N Provos & Honeyman, 2001) tested Stegdetect tool on two million images linked to eBay auctions and showed that over 1% of

the total images appear to have hidden content. However, their study did not show all the results and the details of the testing process are unclear. This research provides the results with all details in simplified tables, taking every result into consideration and analysis. The researcher believes that this is the first example of such detailed study.

6.4 Digital Forensics Investigation

A wide range of criminal investigations use digital evidence that points to a crime, leads to some investigation, supports witness statements or disproves them. Computer or digital forensics in its simplest definition, derived from (Carrier, 2002), refers to the science of recovering materials or data found in digital media to be used as digital evidence for further investigations, especially in relation to computer-related crimes.

Nowadays steganalysis is considered as an important and essential tool to law enforcement, especially in cybercrime and copyright-related cases (Fridrich & Goljan, 2002). However, as it hides information in plain sight, it made a big challenge for law enforcement to detect the existence of hidden content in digital images through visual examination (Craiger et al., 2005). There are several automated steganalysis tools, but these should be used carefully by forensic analyst as they are not reliably accurate.

As stated by (Reith, Carr, & Gunsch, 2002), the methods of obtaining reliable and analysed evidence should be well proved. Thus, the rate of false positives in any tool should be known at the beginning of the investigation process, otherwise there would be a biased investigation and potentially catastrophic results.

(Orebaugh, 2004) tested Stegdetect with 100 images from a digital camera and got 6% false positive rate in their study, whereby all the images were clean, and all detection methods were jphide content.

6.5 Methodology

Stegdetect is chosen for analysis to study the false positive rate aiming to help digital forensics analysts who want to investigate analysing a bulk of digital images. For that purpose, Stegdetect0.6-4 is installed as a Debian package on an Ubunto11.10 operating system running on a laptop with 2.10 GHz Intel Core2 Duo processor and 3 GB of RAM. Also, more than 40000 random image files were downloaded from Google images with Multi Image Downloader (version 1.5.8.4) and tested by Stegdetect with different sensitivity values in the range of (0.1 – 10). In this

study, it is assumed that almost all downloaded images are clean due to the randomness in selection and variation of the source. Additionally, 25000 images are downloaded from the ASIRRA pet images in a compressed folder.

6.6 Finding and Downloading of Images

In this study, the most popular search engine (Google Images) was used to collect more random images with no restrictions to a particular website. The process of searching and downloading of images was undertaken from 9th-13th of February 2012 using Google's Advanced Image Search. The process started first by searching for single English letters (a, b, c… z) and then some common keywords (nature, people, sport, animal, computer, technology, cars and jpg). The resultant images are downloaded by feeding the search's URL to the Multi Image Downloader. The Multi Image Downloader downloads images after refining the URL, adding the start parameter and getting image links. The following are two examples of the search URLs with a single letter 'a' where we turned Safe Search option on and off, respectively.

- http://www.google.com/search?tbm=isch&um=1&hl=en&biw=1366&bih=673&cr=&safe=images&q=a&tbs=ift:jpg
- http://www.google.com/search?tbm=isch&hl=en&biw=1366&bih=673&gbv=2&cr=&safe=off&q=a&tbs=ift:jpg

The purpose behind turning the Safe Search on and off with the same keywords is to get two close, but not identical, sets of images. This will help us to analyse the difference in detection rates between close groups and different ones.

After downloading all image files, the duplicated and some non-jpg images were filtered out to make the results more reliable and robust. This was done for both cases of Safe Search options (on and off).

All other parameters stayed unchanged as shown below:

- Image attribute:
    - Image size: Any
    - Aspect ratio: Any
    - Type of image: Any

- Source of image: Any
- Colour in image: Any

- Usage rights: All images, regardless of license labelling.
- File type: JPG files
- Region: Any region

The other group of images, ASIRRA pet images, were downloaded in a compressed folder from the link (ftp://research.microsoft.com/pub/asirra/petimages.tar) on 11th of June 2012.

6.7 Results

After analysing and recording the results of all 40,303 random images from Google Images, the detection results are distinguished according to the sensitivity value for further investigations on their detection rate. Additionally, it is noticed from the two groups of image results that no significant difference was affected by enabling or disabling Safe Search. The values from the above mentioned groups are all summed up and presented as one overall result. The raw data and other figures of the analysis can be seen in Appendices A and B.

Sensitivity independent results including error, appended, alpha-channel, camouflage, false positive likely, jsteg and f5 stayed unchanged during the analysis with different sensitivity values, as shown in Table 6.1. As mentioned earlier, the stars indicate the level of confidence in detection.

Table 6.1: The rate of sensitivity independent results of 40303 images from Google

Sensitivity	Error	appended	Alpha-channel	Camouflage	Skipped (false positive likely)	jsteg			f5		
						(*)	(**)	(***)	(*)	(**)	(***)
0.1-10	3.16%	0.76%	0.01%	0.02%	10.76%	0.02%	0.00%	0.00%	0.00%	0.00%	0.01%

The errors are the cases where Stegdetect could not analyse the image because of the image format incompatibility (for example, non-RGB images). The highest ratio from the sensitivity

independent results is for 'false positive likely', which is quite high at 10.76%. Other results were low, and nothing special was noted for further discussion.

Sensitivity dependent results including negative, jphide, and outguess(old) were affected by the sensitivity value. There were changes in the level of confidence as well for jphide and outguess(old), as shown in Table 6.2.

Table 6.2: Sensitivity dependent results of 40303 images from Google

Sensitivity	negative	jphide			outguess(old)		
		(*)	(**)	(***)	(*)	(**)	(***)
0.1	84.80%	0.25%	0.03%	0.00%	0.14%	0.06%	0.03%
0.2	83.73%	0.87%	0.22%	0.07%	0.21%	0.08%	0.16%
0.4	82.19%	1.35%	0.56%	0.59%	0.19%	0.12%	0.33%
0.8	78.80%	3.17%	0.88%	1.63%	0.23%	0.10%	0.54%
1.0	**77.41%**	**3.80%**	**0.88%**	**2.08%**	**0.24%**	**0.13%**	**0.57%**
1.6	69.55%	9.01%	2.17%	3.52%	0.34%	0.14%	0.72%
3.2	50.52%	19.20%	6.65%	8.05%	0.21%	0.23%	0.97%
6.4	32.29%	18.63%	11.00%	22.90%	0.02%	0.02%	1.39%
10	26.90%	6.41%	17.64%	33.96%	0.01%	0.01%	1.41%

Negative results were high (84.8%) at the beginning, with low value of sensitivity parameter (0.1) and a gradual decrease between (0.1 – 1.0), then it decreased dramatically between (1.0 – 6.4) and went back to its normal decrease ratio afterwards. This means that the tool is more sensitive in detecting hidden content between (1.0 – 6.4) sensitivity, as shown in Figure 6.1.

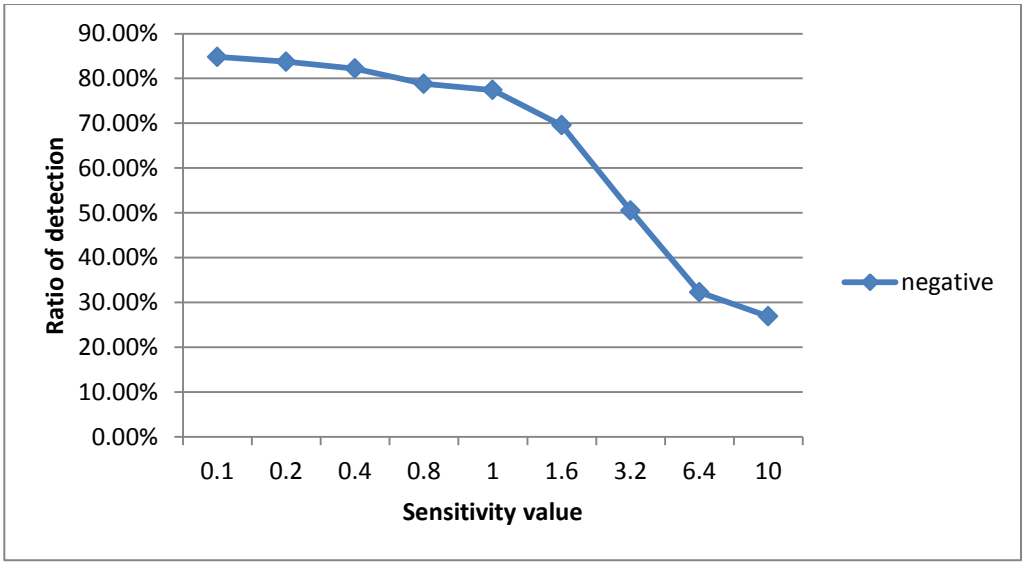


Figure 6.1: Changes in negative ratio with sensitivity value

There is a slight change in jphide results between (0.1 – 1.0), as shown in Figure 6.2. The overall detection of jphide (*, **, ***) increased very much between (1.0 – 3.2). For jphide(**) the rate of change was stable up to (10) and jphide(*) was stable between (3.2 – 6.4), then this goes down afterwards. On the other hand, jphide(***) remains on its rapidly increasing ratio.

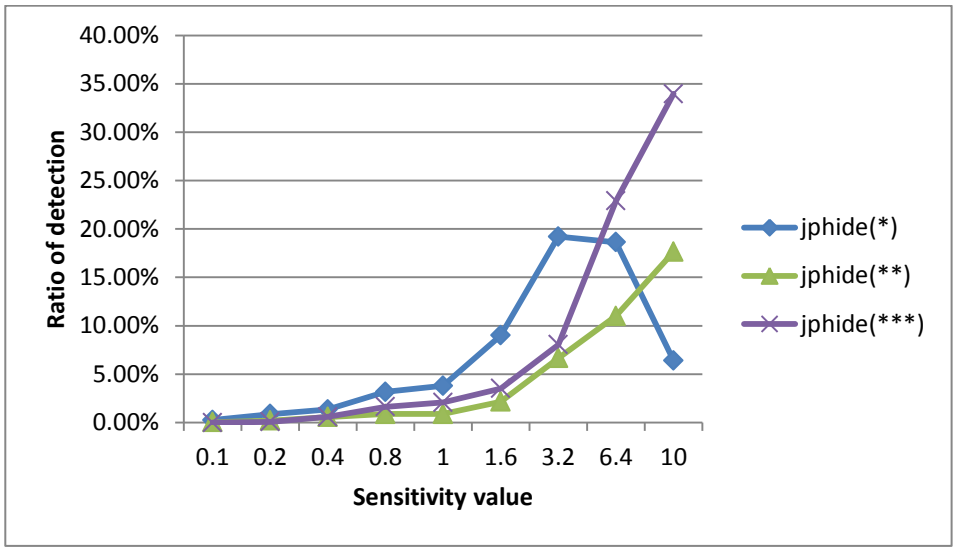


Figure 6.2: Changes in jphide ratio with sensitivity value

From the above graph description we can conclude that the level of confidence is increasing directly with the value of sensitivity, and there is a great increase in overall detection confidence between the sensitivity values (3.2 – 10).

Outguess results were different; the outguess(old)(*) increased between (0.1 – 1.6) and fell down between (1.6 – 6.4) while outguess(old)(**) increased between (0.1 – 3.2) and then fell down afterwards. Finally, outguess(old)(***) increased rapidly between (0.1 – 6.4) and the overall outguess(old) nearly became stable between (6.4 – 10), as shown in Figure 6.3.

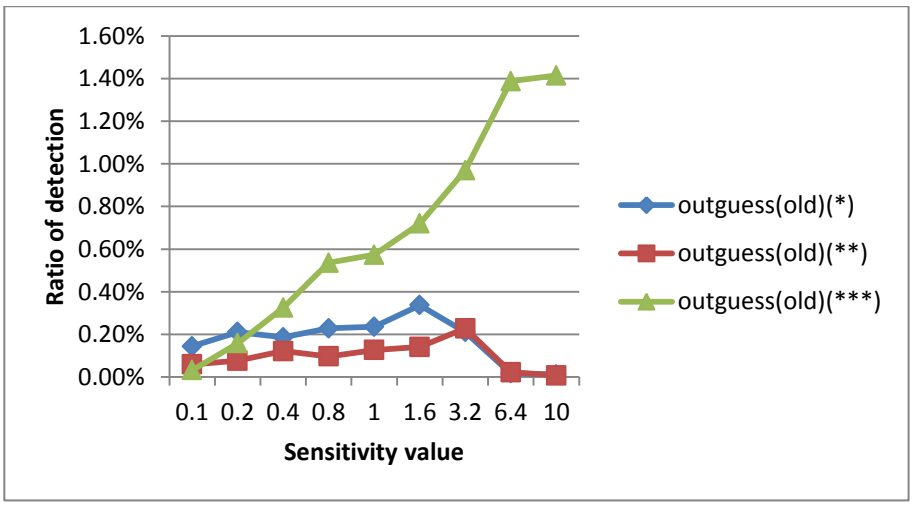


Figure 6.3: Changes in outguess(old) ratio with sensitivity value

In general, the level of detection confidence quickly increases between (0.1 – 6.4) and it almost stabilizes between the sensitivity values (6.4 – 10).

Detecting multiple methods of steganography in the same image yielded one of the interesting results in relation to the change in sensitivity value, as shown in Table 6.3. Table 6.4 shows some images where multiple methods of steganography were detected.

Table 6.3: Examples of detecting multi-methods of steganography

Sensitivity	No. of images	Detected steganographic methods
0.1	27	Appended + false positive likely
	1	F5(***) + false positive likely
0.8	27	Appended + false positive likely
	1	F5(***) + false positive likely
	2	Jphide(*) + appended
	1	Jphide(*) + outguess(old)(***)
	1	Jphide(**) + appended
	1	Jphide(**) + outguess(old)(*)
	2	Jphide(***) + appended

Table 6.4: Examples of detecting multi-methods of steganography

Sensitivity	Detection result
0.1	appended(575)<[nonrandom][data][.....JFIF.....]>
0.2	appended(575)<[nonrandom][data][.....JFIF.....]>
0.4	appended(575)<[nonrandom][data][.....JFIF.....]>
0.8	appended(575)<[nonrandom][data][.....JFIF.....]>
1.0	appended(575)<[nonrandom][data][.....JFIF.....]>
1.6	outguess(old)(*) appended(575)<[nonrandom][data][.....JFIF.....]>
3.2	outguess(old)(**) appended(575)<[nonrandom][data][.....JFIF.....]>
6.4	outguess(old)(***) jphide(*) appended(575)<[nonrandom][data][.....JFIF.....]>
10	outguess(old)(***) jphide(**) appended(575)<[nonrandom][data][.....JFIF.....]>

Sensitivity	Detection result
0.1	negative
0.2	negative
0.4	negative
0.8	negative
1.0	negative
1.6	negative

	3.2	outguess(old)(*) jphide(*)
	6.4	outguess(old)(***) jphide(**)
	10	outguess(old)(***) jphide(***)
	Sensitivity	Detection result
	0.1	negative
	0.2	negative
	0.4	outguess(old)(*)
	0.8	outguess(old)(***) jphide(*)
	1.0	outguess(old)(***) jphide(*)
	1.6	outguess(old)(***) jphide(**)
	3.2	outguess(old)(***) jphide(***)
	6.4	outguess(old)(***) jphide(***)
	10	outguess(old)(***) jphide(***)

To simplify the results of detecting multi-methods of steganography, only the relation between the sensitivity value and the ratio of detecting multi-methods of steganography is shown in Figure 6.5.

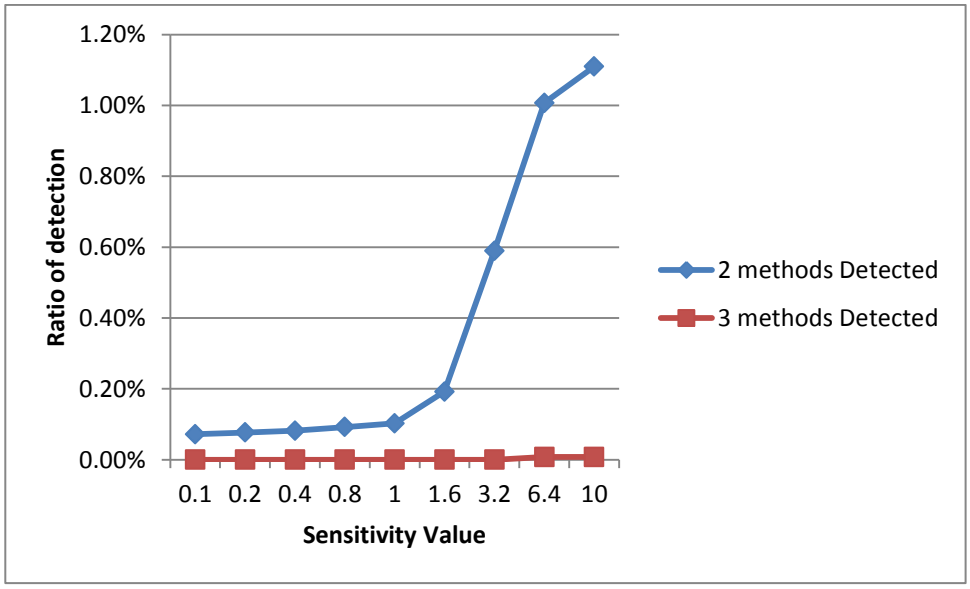


Figure 6.4: The detection ratio of multi-methods of steganography

It is noticeable that the sensitivity value directly affects the detection of multi-methods of steganography, especially two-methods of steganography for sensitivity values (1.6 – 6.4).

Considering all downloaded images as clean is not very accurate due to the possibility of having watermarked images. Nonetheless, the overall false positive rate is considered to be high even after excluding 'errors' and the 'false positives' considered by the tool itself, especially between the sensitivity values of (1.0 – 10). Moreover, the highest rate of false positives comes from jphide

with different levels of confidence. However, the overall false positive rate, in the worst case (sensitivity = 10.0) excluding the jphide, reaches 2.25%, which is much lower than jphide-only ratio (58.01%). This result benefits digital forensics analysts when examining bulk images, when this high rate of false positives should be taken into account for further investigations. Figure 6.5 clarifies the overall picture of the false positive rate for Stegdetect.

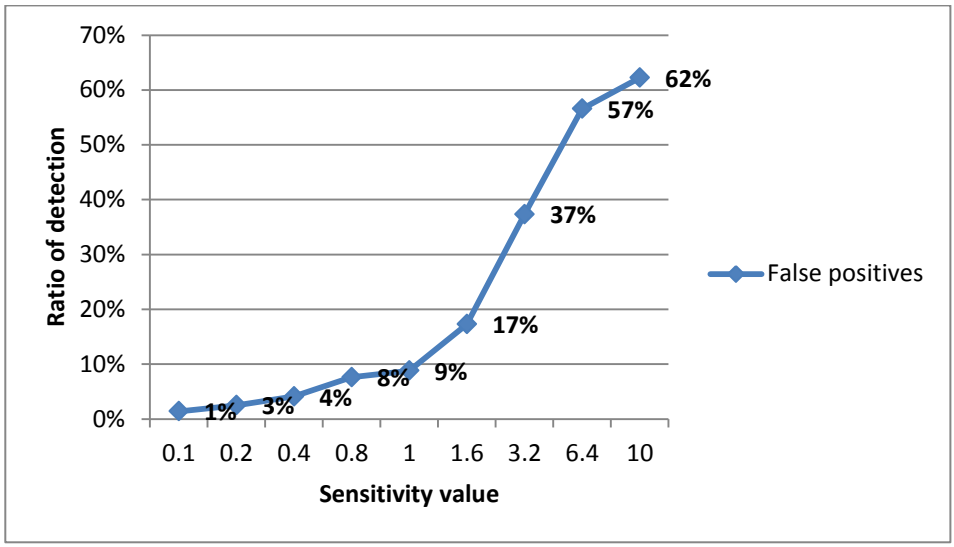


Figure 6.5: The overall false positive ratio

For the other group of images, ASIRRA pet images (cat and dog), the ratio for error, appended, alpha-channel, camouflage, false positive likely, jsteg, and f5 stayed unchanged during the analysis with different sensitivity values, as shown in Table 6.5.

Table 6.5: The ratio of sensitivity independent results of 25000 images from ASIRRA pets

Sensitivity	Error	Appended	Alpha-channel	Camouflage	Skipped (false positive likely)	jsteg			f5		
						(*)	(**)	(***)	(*)	(**)	(***)
0.1-10	0.96 %	0.08 %	0.35 %	0.00 %	3.50 %	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %	0.00 %

Again, the highest ratio from the sensitivity independent results was in the case of false positive likely, which is 3.5%. Other results were low and nothing special exists to be discussed.

The ratio of negative, jphide, and outguess(old) were changed according to the sensitivity value and there were changes in the level of confidence for the cases of jphide and outguess(old), as shown in Table 6.6.

Table 6.6: Sensitivity dependent results of 25000 images from ASIRRA pets

Sensitivity	Negative	jphide			outguess(old)		
		(*)	(**)	(***)	(*)	(**)	(***)
0.1	94.26%	0.54%	0.04%	0.01%	0.16%	0.04%	0.04%
0.2	91.42%	2.70%	0.44%	0.16%	0.16%	0.11%	0.14%
0.4	88.20%	3.13%	1.97%	1.34%	0.11%	0.09%	0.32%
0.8	85.46%	2.61%	1.59%	4.85%	0.16%	0.06%	0.46%
1.0	**83.72%**	**4.91%**	**1.29%**	**5.75%**	**0.14%**	**0.10%**	**0.50%**
1.6	70.86%	14.58%	1.81%	7.23%	0.12%	0.07%	0.61%
3.2	37.45%	33.57%	11.35%	12.27%	0.06%	0.05%	0.75%
6.4	21.67%	15.97%	17.76%	39.44%	0.02%	0.00%	0.86%
10	15.08%	7.51%	15.06%	57.30%	0.01%	0.01%	0.86%

The graphs of the sensitivity-dependent results were very similar to the ones we got from Google Images in both shape and rate of change perspectives. However, there is a slight difference between ratios of detection. The graphs are shown in Figure 6.6 to Figure 6.9.

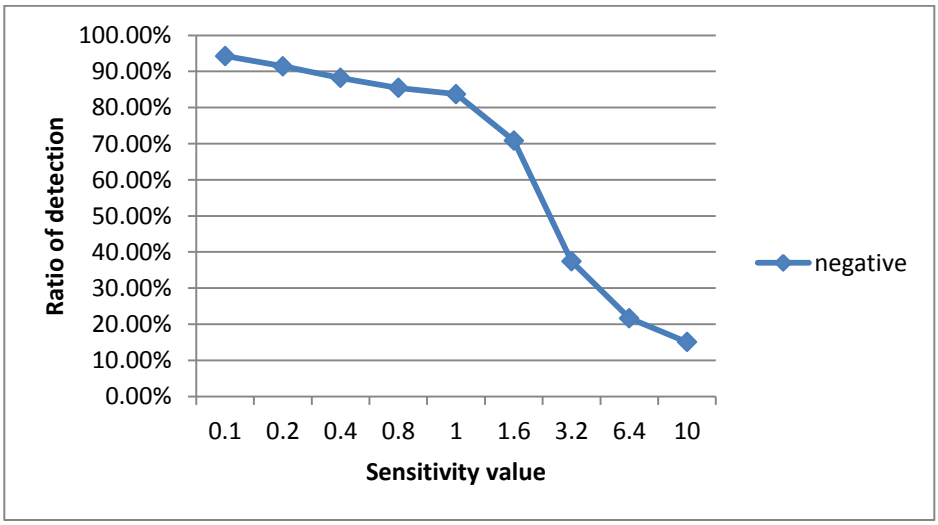


Figure 6.6: Changes in negative ratio with sensitivity value

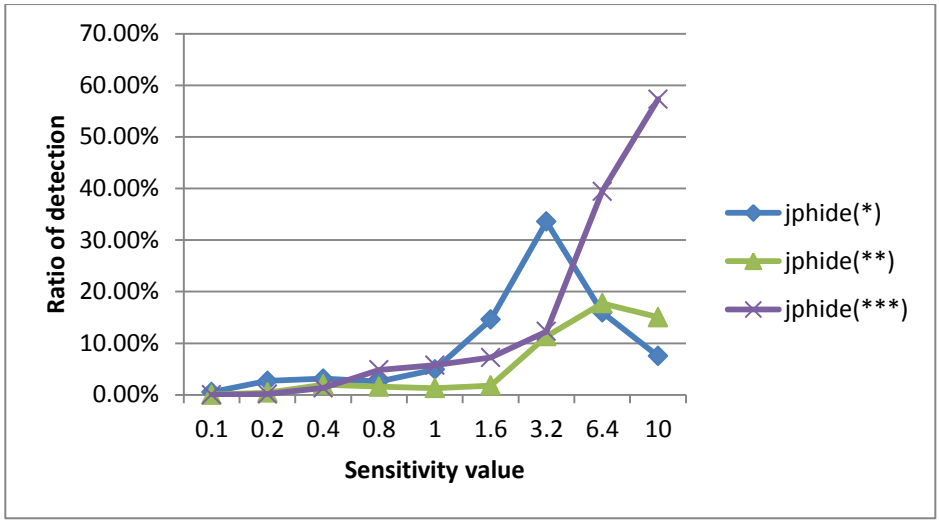


Figure 6.7: Changes in jphide ratio with sensitivity value

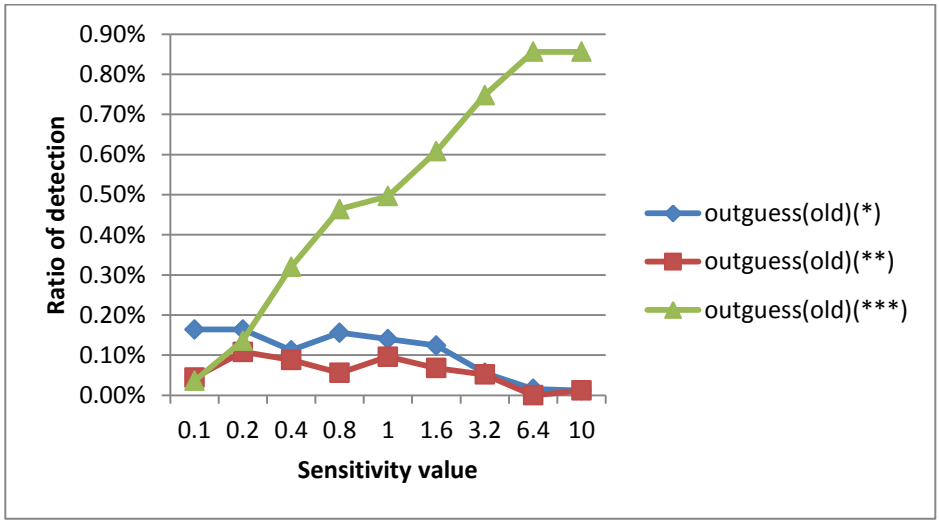


Figure 6.8: Changes in outguess (old) ratio with sensitivity value

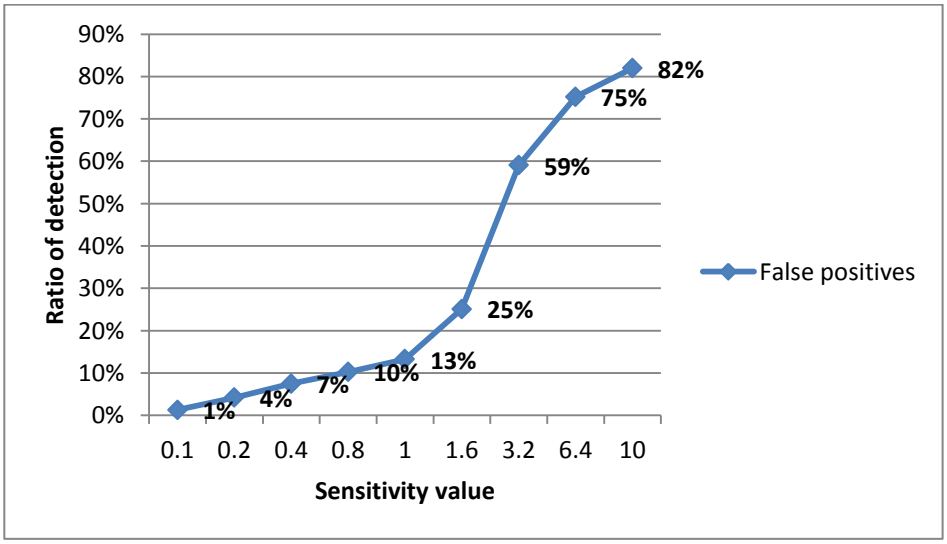


Figure 6.9: The overall false positive ratio

6.8 Statistical Analysis

Assessing and evaluating the accuracy of steganalysis tools and the reliability of their results is not easy, especially for digital forensics analysts. Doing such work involves good knowledge in steganalysis methods, which does not seem to interest most forensic analysts, as they use steganalysis tools as a black box. Providing a simplified method of statistical analysis would thus be very useful for assessing the accuracy of steganalysis tools. It also provides an opportunity to move the evaluation process from 'in laboratory' to the 'real-world' conditions, by considering a random set of images as a baseline of comparison without limiting its features. As mentioned earlier, the random look of digital media files could be assumed as a good indicator to prove the nonexistence of hidden communication.

To study the difference between detection results obtained so far, a statistical method called two-proportion z-test was used to test the hypothesis: 'the two samples are identical'. The two-proportion z-test is used to examine whether two groups differ significantly on some single characteristic. This hypothesis test requires the definition of both a null and an alternative hypothesis.

Since in our case the null hypothesis H_0 states that there is no difference between the two detection proportions, the alternative hypothesis H_a is that there is a difference.

H_0: $p_1=p_2$

H_a: $p_1 \neq p_2$

To get the p-value, which is the probability of observing a sample statistic as extreme as the test statistic, these steps are taken:

- Since the null hypothesis is that $p_1=p_2$, the pooled sample proportion (p) is used to compute the standard error of the sampling distribution:

$$p = \frac{p_1 \times n_1 + p_2 \times n_2}{n_1 + n_2} \qquad \textbf{(6.1)}$$

  Where p_1 and p_2 represent the sample proportion from population 1 and 2 respectively, and n_1 and n_2 are the size of samples 1 and 2 respectively.

- The standard error (SE) of the detection distribution difference between two proportions is computed by the following:

$$SE = \sqrt{p \times (1-p) \times \left[\frac{1}{n_1} + \frac{1}{n_2}\right]} \qquad \textbf{(6.2)}$$

  Where p is the pooled sample proportion.

- Then, the z-score (z), or the test statistic, is calculated by the following equation:

$$z = \frac{p_1 + p_2}{SE} \qquad \textbf{(6.3)}$$

  Where SE is the standard error of the sample distribution.
- As the z-score is used as a test statistic, the normal distribution is used to evaluate the z-score associated probability.

- Finally, the interpretation of the results is given based on the comparison between the p-value and the significant level; the null hypothesis would be rejected if the p-value was less than the significant level.

The significant level is set to 0.05; in this case an error rate of 5% is accepted. Here, the p-value (the probability associated with the z-score) will be computed and compared with the significant level. If the p-value is less than 0.05, the null hypothesis would be rejected; i.e. there is a difference between the proportions of detection results, otherwise they would be considered as identical.

According to the resulting p-value, the significance of the difference in detection proportions can be denoted as follows:

Significant: p-value <0.05

Non-Significant: p-value ≥ 0.05

A statistical test was applied for the two sets of images; the results are shown in Table 6.7 and Table 6.8. The non-significant p-values are coloured with green and the significant ones with red. There are some cells with not applicable (N/A), resulting from having the value of zero from both results (Off and On), which is also coloured with green as there is no significant difference.

The two groups of images from Google with Safe Search option (Off and On) were taken for the test and resulted in only 0.617% (1/162) of red cells, which is far less than 5%, as shown in Table 6.7.

Table 6.7: The difference of detection between Safe Search (Off and On) images

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
							(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
0.1	0.725	0.090	0.678	0.740	0.745	0.773	0.750	0.458	N/A	0.797	0.798	0.572	0.798	0.482	0.486	N/A	N/A	0.798
0.2	0.725	0.090	0.678	0.740	0.745	0.662	0.248	0.767	0.530	0.789	0.788	0.780	0.798	0.482	0.486	N/A	N/A	0.798
0.4	0.725	0.090	0.678	0.740	0.745	0.773	0.542	0.452	0.345	0.660	0.786	0.780	0.798	0.482	0.486	N/A	N/A	0.798
0.8	0.725	0.090	0.678	0.740	0.745	0.710	0.219	0.590	0.396	0.773	0.784	0.782	0.798	0.482	0.486	N/A	N/A	0.798
1	**0.725**	**0.090**	**0.678**	**0.740**	**0.745**	**0.654**	**0.417**	**0.002**	**0.506**	**0.750**	**0.786**	**0.770**	**0.798**	**0.482**	**0.486**	**N/A**	**N/A**	**0.798**
1.6	0.725	0.090	0.678	0.740	0.745	0.388	0.289	0.392	0.787	0.576	0.626	0.796	0.798	0.482	0.486	N/A	N/A	0.798
3.2	0.725	0.090	0.678	0.740	0.745	0.191	0.497	0.354	0.443	0.746	0.483	0.753	0.798	0.482	0.486	N/A	N/A	0.798
6.4	0.725	0.090	0.678	0.740	0.745	0.105	0.764	0.517	0.195	0.740	0.751	0.798	0.798	0.482	0.486	N/A	N/A	0.798
10	0.725	0.090	0.678	0.740	0.745	0.107	0.759	0.730	0.093	0.719	0.672	0.798	0.798	0.482	0.486	N/A	N/A	0.798

This shows that the two groups have similar detection proportions and no significant differences were found. It also implies the acceptance of the null hypothesis ($p_1=p_2$), therefore a digital forensics analyst should not be worried about these two groups of images.

For further investigations, the ASIRRA pet images are taken to perform the same test between the cat and dog images. The obtained results show that 20.37% (33/162) of red cells are obtained, which rejects the null hypothesis ($p_1 \neq p_2$). The red cells result from error, negative, and jphide as shown in Table 6.8. Another important point here is that a clear difference was detected between both groups of ASIRRA image sets (cat and dog), despite having the same source.

Table 6.8: The difference of detection between ASIRRA (cat and dog) images

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
							(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
0.1	0.016	0.161	0.691	0.484	0.535	0.017	0.002	0.086	0.294	0.788	0.762	0.484	0.484	0.484	N/A	N/A	N/A	N/A
0.2	0.016	0.161	0.691	0.484	0.535	0.001	0.052	0.029	0.005	0.788	0.783	0.630	0.484	0.484	N/A	N/A	N/A	N/A
0.4	0.016	0.161	0.691	0.484	0.535	0.007	0.796	0.067	0.005	0.743	0.352	0.323	0.484	0.484	N/A	N/A	N/A	N/A
0.8	0.016	0.161	0.691	0.484	0.535	0.181	0.011	0.138	0.000	0.091	0.798	0.605	0.484	0.484	N/A	N/A	N/A	N/A
1	0.016	0.161	0.691	0.484	0.535	0.765	0.001	0.177	0.001	0.787	0.572	0.690	0.484	0.484	N/A	N/A	N/A	N/A
1.6	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.001	0.022	0.690	0.612	0.787	0.484	0.484	N/A	N/A	N/A	N/A
3.2	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.000	0.084	0.450	0.768	0.779	0.484	0.484	N/A	N/A	N/A	N/A
6.4	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.000	0.254	0.484	N/A	0.733	0.484	0.484	N/A	N/A	N/A	N/A
10	0.016	0.161	0.691	0.484	0.535	0.548	0.000	0.001	0.000	0.675	0.675	0.733	0.484	0.484	N/A	N/A	N/A	N/A

The digital forensics analyst will benefit from the results as they indicate the area of differences for further investigation process. Here, the detection results of error, negative, and jphide may be considered for further study by the digital forensics analyst. Certain image processing and filtering techniques may have been applied before publishing the ASIRRA pet images, which also should be considered by the digital forensics analyst.

To test the randomness of the ASIRRA image set, the random set of images from Google images is considered as a baseline for comparison. The results show that there is a remarkable difference in detection between these two sets, as there are 79 red cells among the total of 162 cells, comprising more than 48% of the total. Hence, the null hypothesis is rejected, proving that the ASIRRA images do not have a random look, which reflects the reality, as shown in Table 6.9.

Table 6.9: The difference of detection between random Google and ASIRRA images

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
							(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
0.1	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.671	0.159	0.650	0.563	0.773	0.196	0.753	0.585	N/A	N/A	0.124
0.2	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.003	0.331	0.344	0.611	0.196	0.753	0.585	N/A	N/A	0.124
0.4	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.054	0.358	0.793	0.196	0.753	0.585	N/A	N/A	0.124
0.8	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.107	0.164	0.362	0.196	0.753	0.585	N/A	N/A	0.124
1	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.023	0.426	0.341	0.196	0.753	0.585	N/A	N/A	0.124
1.6	0.000	0.000	0.000	0.259	0.000	0.001	0.000	0.006	0.000	0.000	0.020	0.191	0.196	0.753	0.585	N/A	N/A	0.124
3.2	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.010	0.196	0.753	0.585	N/A	N/A	0.124
6.4	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.791	0.049	0.000	0.196	0.753	0.585	N/A	N/A	0.124
10	0.000	0.000	0.000	0.259	0.000	0.000	0.000	0.000	0.000	0.797	0.670	0.000	0.196	0.753	0.585	N/A	N/A	0.124

Another very interesting application is that it could be used to show the similarity between two sets of images after getting their feature representation, to avoid the cover source mismatch in the evaluation process of any steganalysis tool. The proposed statistical method can also be used to evaluate the steganalysis tools by applying two different detection methods on the same set of images and analysing their detection results to show the area of differences between them.

6.9 Conclusion

In this study, we analysed one of the well-known digital image steganalysis tools (Stegdetect) to examine its false positive rates. This study could benefit digital forensics analysts in their investigations. We concluded that the value of the sensitivity parameter strongly affects the detection rate for jphide and outguess(old), especially when the sensitivity value is between (1.0 –

6.4). Another conclusion, possibly the more important one, is that we have noticed a high rate of false positives, particularly between sensitivity values of (1.0 – 10). For this reason, we can indicate the sensitivity value of 1.0 as an optimum value for detection, as the detection of 'negative' sharply falls down after this point. This high rate of false positives should be taken into consideration by digital forensics analysts when processing, as is frequently the case, large numbers of images during an investigation using Stegdetect.

Finally, a very useful statistical method has been proposed to show the differences in proportion of detection between two groups of images in a very simple way. The most random group of images could act as a baseline for this comparison (Google Images in our case). This would help the digital forensics analyst to take further informed decisions during an investigation process, likely arriving at better evidence. This statistical method could be applied to any other steganalysis tools, especially when the analyst has no prior information about the false positive rate of the chosen tool.

The proposed statistical method can also be used to evaluate the steganalysis methods (tools) by applying two different detection methods on the same set of images and statistically analysing their results. The significant differences can be used as a base to improve a certain steganalysis method.

There are two other related studies that could be addressed in future works: one is based on studying the false negative rate of Stegdetect, the other is applying similar analysis on other steganalysis tools.

CHAPTER 7: CONCLUSIONS AND FUTURE PERSPECTIVES

7.1 Overview

Undetectability represents the most important aspect of any steganographic system. Thus, this thesis considered the undetectability in three related aspects of steganography: embedding, detection, and the analysis of detection results for evaluation and digital forensics investigation process.

This thesis adds value to research and practice communities concerned with data hiding, image steganography, steganalysis and the digital forensics investigation process. The novel methods proposed in these relevant research areas also enhanced the value of contributions made in this research. These contributions are evaluated, peer reviewed, and published in four conference papers and one journal article. It has also been reviewed by a number of other top conferences and journals in the field of information hiding, steganography, steganalysis and digital investigation.

This chapter discusses the conclusions and the future perspectives of this thesis by describing the research findings, limitations, and future research directions.

7.2 Research Findings

The contributions and the research findings of this thesis are discussed under three different but strongly related domains. The first important finding is about the novel steganography method, proposed in chapter four, that was applied in both LSB and 2LSB image steganography in such a way that it improved embedding efficiency. Thus, as the embedding efficiency directly affects the probability of detection, this novel approach significantly reduced the probability of detection, which is about 40% of the ordinary LSB and 2LSB steganography methods. The other improvement is that the proposed method reduced the bit-level cost of change to the cover image for the same message length, which would again reduce the probability of detection by binary similarity measure steganalysis methods. Moreover, the proposed embedding method could be applied for the embedding rate of 1 with no skipping of saturated pixel values (0 and 255).

The second important finding is about the detection method of the 2LSB image steganography discussed in chapter five. In addition to the high accuracy in detection, the discrete version of the classifier can give labels to the analysed images instead of the probability of having hidden

contents. This method eliminates the overhead of choosing the right threshold value for classification; hence, it could be used as an automated tool for classifying a bulk of images by inexperienced people in the field of steganalysis. The probabilistic version of the proposed detection method is also very accurate, and outperforms the current 2LSB steganalysis methods. This is due to the fact that the proposed method relies on some measurements that stay unmodified before and after the embedding process has taken place. Consequently, it could maintain its accuracy in detection for low embedding rates.

For the evaluation of the steganalysis tools and the digital forensics investigation process, this research proposed a statistical method that could be applied on the detection results of the steganalysis tools, as discussed in chapter six. It could be used to evaluate the steganalysis tools without the need to have detailed knowledge about the detection method; in other words, for users who are using the detection tools as a black box. This could be achieved by applying and comparing the results of more than one steganalysis tools on the same set of images.

The other usage of the proposed statistical method is to apply a certain steganalysis tool on two different image sets for investigation by the digital forensics analyst. The first set could be a random set of images that can act as a baseline for comparison, and the other set would be the testing set. Hence, the significant areas of differences between the two sets of images can be extracted from the detection results that let the digital forensics analyst do more investigation on those significant areas and neglect the insignificant ones. In this case, it reduces the cost, complexity, and time duration needed by the investigation process.

7.3 Research Limitations

In spite of having many contributions and additions to the current knowledge of steganography and steganalysis, there are also some limitations that can be considered in future researches. The proposed embedding method, single mismatch, adds a little distortion to the stego image and produces a lower PSNR value compared to other embedding methods discussed in chapter four. This extra distortion, from the fidelity point of view, resulted from involving few higher bit planes in the embedding process. However, this difference in PSNR value is only 1.75 dB for SMLSB and 3.8 dB for SM2LSB that can be tolerated, as the PSNR values are very close to the other methods and very far from the lower limit value. Another limitation of the proposed embedding method is that it cannot totally defeat the detection methods, which is the case for every steganographic method, due to the modifications in the cover image pixel values.

The targeted image steganalysis methods usually consider the characteristics of the embedding process and its effects on the image pixel value transitions to detect the availability of hidden contents. The proposed 2LSB detection method (EPoV), just like other targeted steganalysis methods, has a limitation of differentiating the noise from the message embedding, when they cause the same changing pattern on the cover image. However, most steganographic references define the embedding process as adding noise to the cover media.

The evaluation process of steganalysis methods (tools) and the digital forensics investigation process are both simplified by the proposed statistical method described in chapter six. However, it can only be applied on two sets of images. Hence, the inability to use three or more image sets could be counted as a limitation of the proposed method. Another limitation is that the proposed method does not consider the features of the image set used as a baseline for comparison, thus it would be possible to have a huge difference between both image sets in characteristics and causes of 'source mismatch', which could mislead the analyst during the digital forensics investigation process.

7.4 Future Research

As mentioned before, this thesis is built on a three-fold research, and future perspectives could also be discussed in three directions. The first one is about the proposed embedding method, single mismatch LSB and 2LSB steganography, which might be improved in two different ways. First, there is a possibility to develop some pre-processing methods on the secret message in such a way that needs less change during the embedding process. Secondly, it might be possible to modify the embedding method to produce a higher value of PSNR without affecting the reduced probability of detection.

The second future perspective is about the proposed detection method, EPoV, for detecting 2LSB image steganography. It might be possible to improve the detection accuracy of the proposed method by applying other statistical methods. Also, there is an opportunity to use this detection method for attacking some publicly available steganography applications as a testing tool, especially the discrete classifier. In addition to the possibility of improving the detection accuracy, differentiating the noise insertion from the 2LSB data embedding could also be subject of further research, possibly considering some features of the image that could be affected differently by the embedding and noise insertion process.

The last future perspective is about the proposed statistical method for evaluating steganalysis tools and simplifying the investigation process by the digital forensics analyst. There is a possibility to extend the statistical method to be applied on more than two sets of images and to specify the significant areas of differences based on some other parameters. Another possibility is to add some feature selection methods to be applied on the testing images and then choose a suitable set of random images as a baseline for comparison. This would eliminate the source mismatch between the two sets of images, the baseline and the testing image sets, and reduce the probability of misleading the investigation process.

APPENDICES

A. The following tables are the raw results of detection for each group of images.

Table A.1: The detection results of Safe search option (On)

No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
20063	0.1	626	170	1	4	2148	17023	49	5	0	29	12	5	4	1	0	0	0	3
20063	0.2	626	170	1	4	2148	16821	160	42	12	43	15	31	4	1	0	0	0	3
20063	0.4	626	170	1	4	2148	16500	282	105	109	40	25	64	4	1	0	0	0	3
20063	0.8	626	170	1	4	2148	15790	665	184	312	47	20	109	4	1	0	0	0	3
20063	1	626	170	1	4	2148	15504	785	144	403	49	26	117	4	1	0	0	0	3
20063	1.6	626	170	1	4	2148	13898	1849	452	709	63	31	145	4	1	0	0	0	3
20063	3.2	626	170	1	4	2148	10051	3891	1366	1644	44	41	198	4	1	0	0	0	3
20063	6.4	626	170	1	4	2148	6384	3749	2236	4665	4	5	278	4	1	0	0	0	3
20063	10	626	170	1	4	2148	5308	1279	3555	6912	3	2	284	4	1	0	0	0	3

Table A.2: The detection results of Safe search option (Off)

No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
20240	0.1	647	135	2	3	2190	17155	53	9	0	29	12	8	4	0	1	0	0	3
20240	0.2	647	135	2	3	2190	16924	190	45	17	42	16	33	4	0	1	0	0	3
20240	0.4	647	135	2	3	2190	16626	264	122	130	35	24	67	4	0	1	0	0	3
20240	0.8	647	135	2	3	2190	15969	614	171	345	45	19	107	4	0	1	0	0	3
20240	1	647	135	2	3	2190	15694	748	210	434	46	25	114	4	0	1	0	0	3
20240	1.6	647	135	2	3	2190	14132	1783	421	709	73	26	145	4	0	1	0	0	3
20240	3.2	647	135	2	3	2190	10310	3848	1314	1599	41	51	193	4	0	1	0	0	3
20240	6.4	647	135	2	3	2190	6630	3759	2197	4564	3	4	281	4	0	1	0	0	3
20240	10	647	135	2	3	2190	5534	1306	3554	6775	2	1	286	4	0	1	0	0	3

Table A.3: The detection results of ASIRRA pet images (Cat)

No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
12500	0.1	98	14	46	0	425	11834	48	2	0	21	6	6	0	0	0	0	0	0
12500	0.2	98	14	46	0	425	11507	308	42	10	21	14	19	0	0	0	0	0	0
12500	0.4	98	14	46	0	425	11103	390	222	138	15	8	46	0	0	0	0	0	0
12500	0.8	98	14	46	0	425	10731	363	217	533	13	7	62	0	0	0	0	0	0
12500	1	98	14	46	0	425	10457	552	177	651	17	10	65	0	0	0	0	0	0
12500	1.6	98	14	46	0	425	8698	2030	264	849	17	7	75	0	0	0	0	0	0
12500	3.2	98	14	46	0	425	4942	3777	1554	1589	5	7	92	0	0	0	0	0	0
12500	6.4	98	14	46	0	425	2854	2110	1932	4988	3	0	104	0	0	0	0	0	0
12500	10	98	14	46	0	425	1909	1073	1986	6929	2	2	104	0	0	0	0	0	0

Table A.4: The detection results of ASIRRA pet images (Dog)

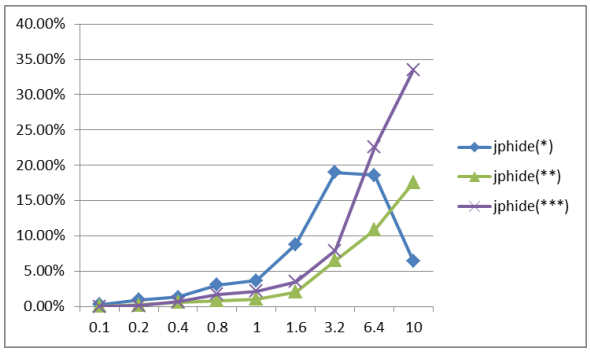
No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
12500	0.1	141	6	41	1	451	11732	88	9	2	20	5	3	1	1	0	0	0	0
12500	0.2	141	6	41	1	451	11347	368	69	30	20	13	15	1	1	0	0	0	0
12500	0.4	141	6	41	1	451	10946	392	271	196	13	14	34	1	1	0	0	0	0
12500	0.8	141	6	41	1	451	10635	289	180	679	26	7	54	1	1	0	0	0	0
12500	1	141	6	41	1	451	10474	675	146	787	18	14	59	1	1	0	0	0	0
12500	1.6	141	6	41	1	451	9016	1615	189	959	14	10	77	1	1	0	0	0	0
12500	3.2	141	6	41	1	451	4421	4615	1284	1479	9	6	95	1	1	0	0	0	0
12500	6.4	141	6	41	1	451	2563	1882	2507	4871	1	0	110	1	1	0	0	0	0
12500	10	141	6	41	1	451	1860	804	1778	7397	1	1	110	1	1	0	0	0	0

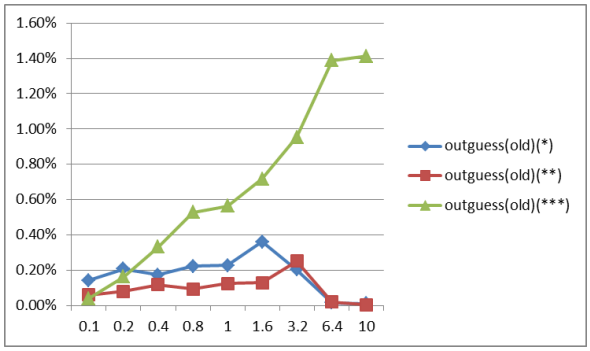
Table A.5: The detection results of ASIRRA pet images

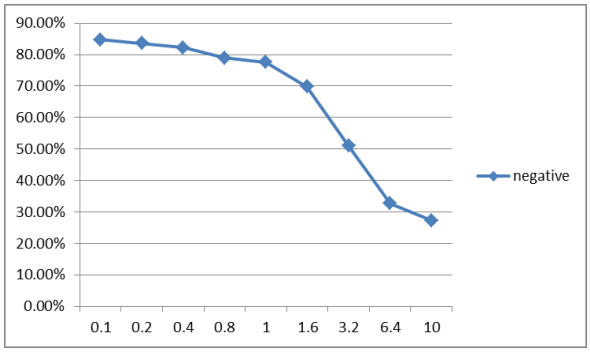
No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
25000	0.1	239	20	87	1	876	23566	136	11	2	41	11	9	1	1	0	0	0	0
25000	0.2	239	20	87	1	876	22854	676	111	40	41	27	34	1	1	0	0	0	0
25000	0.4	239	20	87	1	876	22049	782	493	334	28	22	80	1	1	0	0	0	0
25000	0.8	239	20	87	1	876	21366	652	397	1212	39	14	116	1	1	0	0	0	0
25000	1	239	20	87	1	876	20931	1227	323	1438	35	24	124	1	1	0	0	0	0
25000	1.6	239	20	87	1	876	17714	3645	453	1808	31	17	152	1	1	0	0	0	0
25000	3.2	239	20	87	1	876	9363	8392	2838	3068	14	13	187	1	1	0	0	0	0
25000	6.4	239	20	87	1	876	5417	3992	4439	9859	4	0	214	1	1	0	0	0	0
25000	10	239	20	87	1	876	3769	1877	3764	14326	3	3	214	1	1	0	0	0	0

B. The following graphs are the results of detection rate for each group of images:

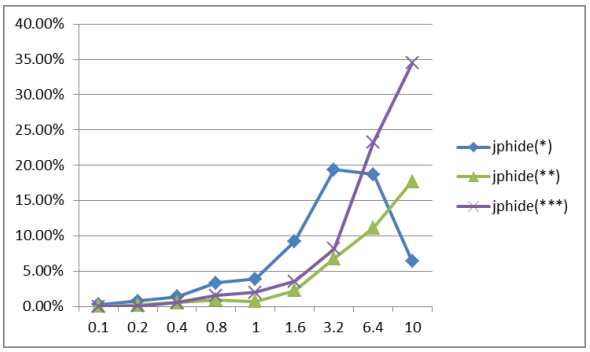
Google images - Safe search option Off

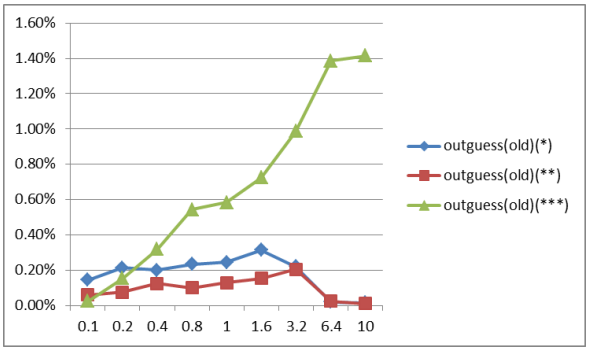


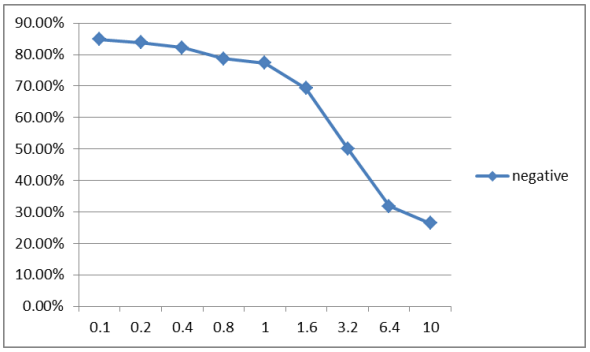




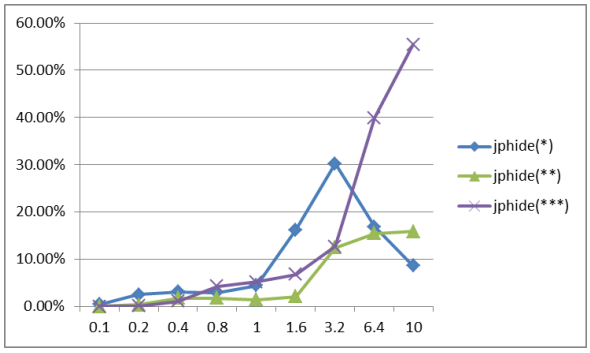
Google images - Safe search option On

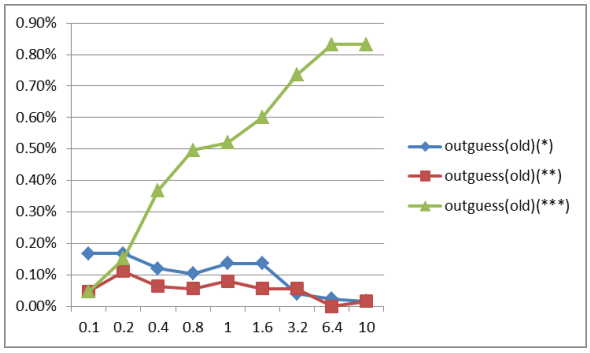


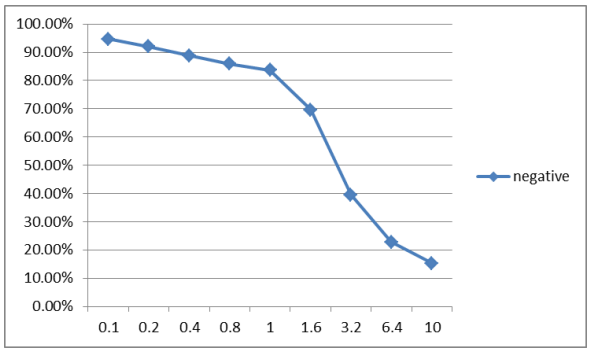




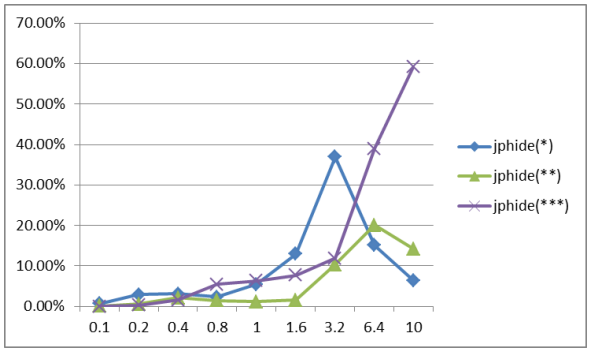
ASIRRA Cat images

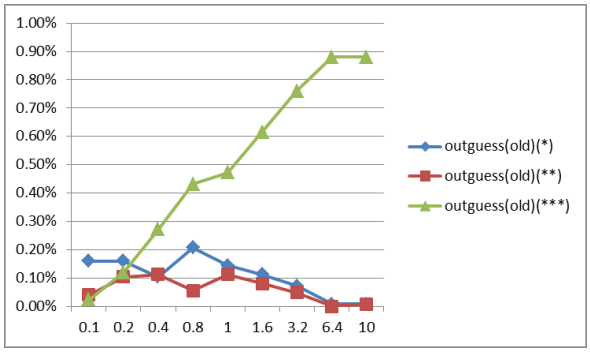


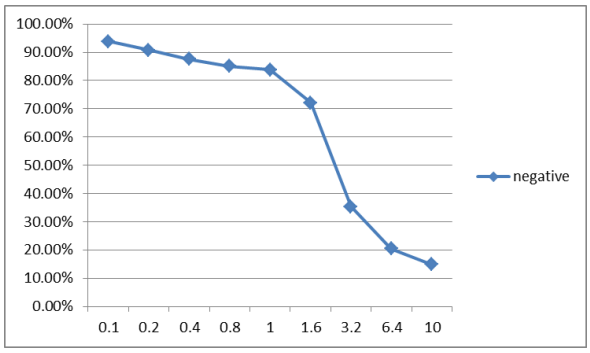




ASIRRA Dog images







REFERENCES

Al-Mohammad, A. (2010). *Steganography-based secret and reliable communications: Improving steganographic capacity and imperceptibility.* Brunel University, School of Information Systems, Computing and Mathematics Theses.

Almohammad, A. (2010). *Steganography-Based Secret and Reliable Communications: Improving Steganographic Capacity and Imperceptibility.* (Doctor of Philosophy), Brunel University.

Alturki, F., & Mersereau, R. (2001, 7-10 Oct 2001). *Secure blind image steganographic technique using discrete Fourier transformation.* Paper presented at the Image Processing, 2001. Proceedings. 2001 International Conference on.

Alvarez, P. (2004). Using extended file information (EXIF) file headers in digital evidence analysis. *International Journal of Digital Evidence, 2*(3), 1-5.

Amin, M. M., Salleh, M., Ibrahim, S., Katmin, M. R., & Shamsuddin, M. Z. I. (2003, 14-15 Jan. 2003). *Information hiding using steganography.* Paper presented at the Telecommunication Technology, 2003. NCTT 2003 Proceedings. 4th National Conference on.

Anand, D., & Niranjan, U. C. (1998, 29 Oct-1 Nov 1998). *Watermarking medical images with patient information.* Paper presented at the Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th Annual International Conference of the IEEE.

Anderson, R. J., & Petitcolas, F. A. (1998). On the limits of steganography. *Selected Areas in Communications, IEEE Journal on, 16*(4), 474-481.

Artz, D. (2001). Digital steganography: hiding data within data. *Internet Computing, IEEE, 5*(3), 75-80. doi: 10.1109/4236.935180

Aura, T. (1996). Practical invisibility in digital communication. In R. Anderson (Ed.), *Information Hiding* (Vol. 1174, pp. 265-278): Springer Berlin Heidelberg.

Avcibaş, I., Kharrazi, M., Memon, N., & Sankur, B. (2005). Image steganalysis with binary similarity measures. *EURASIP J. Appl. Signal Process., 2005*, 2749-2757. doi: 10.1155/asp.2005.2749

Bailey, K., Curran, K., & Condell, J. (2004). Evaluation of pixel-based steganography and stegodetection methods. *Imaging Science Journal, The, 52*(3), 131-150.

Bender, W., Butera, W., Gruhl, D., Hwang, R., Paiz, F. J., & Pogreb, S. (2000). Applications for data hiding. *IBM Systems Journal, 39*(3.4), 547-568. doi: 10.1147/sj.393.0547

Bender, W., Gruhl, D., Morimoto, N., & Lu, A. (1996). Techniques for data hiding. *IBM Systems Journal, 35*(3.4), 313-336. doi: 10.1147/sj.353.0313

Boato, G., Conotter, V., De Natale, F. G., & Fontanari, C. (2009). Watermarking robustness evaluation based on perceptual quality via genetic algorithms. *Information Forensics and Security, IEEE Transactions on, 4*(2), 207-216.

Böhme, R., & Ker, A. D. (2006). *A two-factor error model for quantitative steganalysis.* Paper presented at the Electronic Imaging 2006.

Böhme, R., & Westfeld, A. (2004). Breaking Cauchy Model-Based JPEG Steganography with First Order Statistics. In P. Samarati, P. Ryan, D. Gollmann & R. Molva (Eds.), *Computer Security – ESORICS 2004* (Vol. 3193, pp. 125-140): Springer Berlin Heidelberg.

Buccigrossi, R. W., & Simoncelli, E. P. (1999). Image compression via joint statistical characterization in the wavelet domain. *Image Processing, IEEE Transactions on, 8*(12), 1688-1701. doi: 10.1109/83.806616

Cancelli, G., Doerr, G., Barni, M., & Cox, I. J. (2008, 8-10 Oct. 2008). *A comparative study of ±1 steganalyzers.* Paper presented at the Multimedia Signal Processing, 2008 IEEE 10th Workshop on.

Cancelli, G., Doërr, G., Cox, I. J., & Barni, M. (2008). *Detection of±1 LSB steganography based on the amplitude of histogram local extrema.* Paper presented at the Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on.

Carrier, B. (2002). Defining Digital Forensic Examination and Analysis Tools. *Digital Forensics Research Workshop II*.

Cayre, F., Fontaine, C., & Furon, T. (2005). *Watermarking security part one: Theory*. Paper presented at the Electronic Imaging 2005.

Cayre, F., & Macq, B. (2003). Data hiding on 3-D triangle meshes. *Signal Processing, IEEE Transactions on, 51*(4), 939-949. doi: 10.1109/tsp.2003.809380

Chan, C.-S. (2009). On using LSB matching function for data hiding in pixels. *Fundamenta Informaticae, 96*(1-2), 49-59.

Chandramouli, R. (2002). *Mathematical approach to steganalysis.* Paper presented at the Electronic Imaging 2002.

Chandramouli, R. (2003). A mathematical framework for active steganalysis. *Multimedia Systems, 9*(3), 303-311.

Chandramouli, R., Kharrazi, M., & Memon, N. (2004). Image steganography and steganalysis: Concepts and practice *Digital Watermarking* (pp. 35-49): Springer.

Chandramouli, R., & Memon, N. (2001). *Analysis of LSB based image steganography techniques.* Paper presented at the International Conference on Image Processing 2001.

Chang, C.-C., Chen, T.-S., & Chung, L.-Z. (2002). A steganographic method based upon JPEG and quantization table modification. *Information Sciences, 141*(1–2), 123-138. doi: http://dx.doi.org/10.1016/S0020-0255(01)00194-3

Chang, C.-C., Hu, Y.-S., & Lu, T.-C. (2006). A watermarking-based image ownership and tampering authentication scheme. *Pattern Recognition Letters, 27*(5), 439-446. doi: http://dx.doi.org/10.1016/j.patrec.2005.09.006

Chang, C.-C., Lin, C.-Y., & Wang, Y.-Z. (2006). New image steganographic methods using run-length approach. *Information Sciences, 176*(22), 3393-3408. doi: http://dx.doi.org/10.1016/j.ins.2006.02.008

Chang, C.-C., & Tseng, H.-W. (2004). A steganographic method for digital images using side match. *Pattern Recognition Letters, 25*(12), 1431-1437. doi: http://dx.doi.org/10.1016/j.patrec.2004.05.006

Cheddad, A., Condell, J., Curran, K., & Mc Kevitt, P. (2010). Digital image steganography: Survey and analysis of current methods. *Signal Processing, 90*(3), 727-752. doi: http://dx.doi.org/10.1016/j.sigpro.2009.08.010

Chen, L.-H., & Lee, Y.-K. (2003). Secure Error-Free Steganography for JPEG Images. *International Journal of Pattern Recognition and Artificial Intelligence, 17*(06), 967-981. doi: doi:10.1142/S021800140300268X

Chorein, A. (2008). SilentEye - Steganography is yours. 2014, from http://www.silenteye.org/

Chunhua, C., & Shi, Y. Q. (2008, 18-21 May 2008). *JPEG image steganalysis utilizing both intrablock and interblock correlations.* Paper presented at the Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on.

Chutani, S., & Goyal, H. (2012). LSB Embedding in Spatial Domain-A Review of Improved Techniques. *International Journal of Computers & Technology, 3*(1), 153-157.

Cogranne, R., & Retraint, F. (2013). An Asymptotically Uniformly Most Powerful Test for LSB Matching Detection. *Information Forensics and Security, IEEE Transactions on, 8*(3), 464-476. doi: 10.1109/TIFS.2013.2238232

Cole, E., & Krutz, R. D. (2003). *Hiding in plain sight: Steganography and the art of covert communication*: John Wiley & Sons, Inc.

Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2008). *Digital Watermarking and Steganography*: Morgan Kaufmann Publishers Inc.

Craiger, J. P., Pollitt, M., & Swauger, J. (2005). Law enforcement and digital evidence. *Handbook of Information Security, New York, USA*.

Crandall, R. (1998). Some notes on steganography. *Posted on steganography mailing list*.

Craver, S. (1998). *On public-key steganography in the presence of an active warden.* Paper presented at the Information Hiding.

Dabeer, O., Sullivan, K., Madhow, U., Chandrasekaran, S., & Manjunath, B. S. (2004). Detection of hiding in the least significant bit. *Signal Processing, IEEE Transactions on, 52*(10), 3046-3058. doi: 10.1109/tsp.2004.833869

Douceur, J., Elson, J., & Howell, J. ASIRRA -- Public Corpus. 2013, from http://research.microsoft.com/en-us/projects/asirra/corpus.aspx

Dumitrescu, S., & Wu, X. (2005). *LSB steganalysis based on high-order statistics*. Paper presented at the Proceedings of the 7th workshop on Multimedia and security, New York, NY, USA.

Dumitrescu, S., Wu, X., & Memon, N. (2002). *On steganalysis of random LSB embedding in continuous-tone images.* Paper presented at the Image Processing. 2002. Proceedings. 2002 International Conference on.

Dumitrescu, S., Wu, X., & Wang, Z. (2003). Detection of LSB steganography via sample pair analysis. *Signal Processing, IEEE Transactions on, 51*(7), 1995-2007.

Dunbar, B. (2002). A detailed look at Steganographic Techniques and their use in an Open-Systems Environment. *Sans Institute*.

Embedded, T. (1996). *Information hiding terminology.* Paper presented at the Information Hiding: First International Workshop, Cambridge, UK, May 30-June 1, 1996. Proceedings.

Farid, H. (2002, 2002). *Detecting hidden messages using higher-order statistical models.* Paper presented at the Image Processing. 2002. Proceedings. 2002 International Conference on.

Fawcett, T. (2003). ROC Graphs: Notes and Practical Considerations for Data Mining Researchers. CA: HP Laboratories: HP Laboratories Palo Alto.

Fridrich, J. (2005). Feature-Based Steganalysis for JPEG Images and Its Implications for Future Design of Steganographic Schemes. In J. Fridrich (Ed.), *Information Hiding* (Vol. 3200, pp. 67-81): Springer Berlin Heidelberg.

Fridrich, J. (2009). *Steganography in Digital Media: Principles, Algorithms, and Applications*: Cambridge University Press.

Fridrich, J., & Du, R. (2000). Secure Steganographic Methods for Palette Images. In A. Pfitzmann (Ed.), *Information Hiding* (Vol. 1768, pp. 47-60): Springer Berlin Heidelberg.

Fridrich, J., & Goljan, M. (2002). *Practical steganalysis of digital images: state of the art.* Paper presented at the SPIE Photonics West, Electronic Imaging, CA.

Fridrich, J., & Goljan, M. (2003). *Digital image steganography using stochastic modulation.* Paper presented at the Electronic Imaging 2003.

Fridrich, J., & Goljan, M. (2004). *On estimation of secret message length in LSB steganography in spatial domain.* Paper presented at the Electronic Imaging 2004.

Fridrich, J., Goljan, M., & Du, R. (2001a). Detecting LSB steganography in color, and gray-scale images. *Multimedia, IEEE, 8*(4), 22-28.

Fridrich, J., Goljan, M., & Du, R. (2001b). *Reliable detection of LSB steganography in color and grayscale images*. Paper presented at the Proceedings of the 2001 workshop on Multimedia and security: new challenges, Ottawa, Ontario, Canada.

Fridrich, J., Goljan, M., Hogea, D., & Soukal, D. (2003). Quantitative steganalysis of digital images: estimating the secret message length. *Multimedia Systems, 9*(3), 288-302. doi: 10.1007/s00530-003-0100-9

Fridrich, J., Goljan, M., & Soukal, D. (2003). *Higher-order statistical steganalysis of palette images.* Paper presented at the Security and Watermarking of Multimedia Contents V.

Fridrich, J., Goljan, M., & Soukal, D. (2005). *Steganography via codes for memory with defective cells.* Paper presented at the 43rd Conference on Coding, Communication, and Control.

Fridrich, J., Lisoněk, P., & Soukal, D. (2007). On Steganographic Embedding Efficiency. In J. Camenisch, C. Collberg, N. Johnson & P. Sallee (Eds.), *Information Hiding* (Vol. 4437, pp. 282-296): Springer Berlin Heidelberg.

Fridrich, J., & Long, M. (2000, 2000). *Steganalysis of LSB encoding in color images.* Paper presented at the Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on.

Fridrich, J., Pevný, T., & Kodovský, J. (2007). *Statistically undetectable jpeg steganography: dead ends challenges, and opportunities.* Paper presented at the Proceedings of the 9th workshop on Multimedia & security.

Fridrich, J., & Soukal, D. (2006). *Matrix embedding for large payloads.* Paper presented at the Electronic Imaging 2006.

Fridrich, J., Soukal, D., & Goljan, M. (2005). *Maximum likelihood estimation of length of secret message embedded using±k steganography in spatial domain.* Paper presented at the Electronic Imaging 2005.

Gireesh Kumar, T., Jithin, R., & Shankar, D. D. (2010, 20-21 June 2010). *Feature Based Steganalysis Using Wavelet Decomposition and Magnitude Statistics.* Paper presented at the Advances in Computer Engineering (ACE), 2010 International Conference on.

Goljan, M., Fridrich, J., & Holotyak, T. (2006). *New blind steganalysis and its implications.* Paper presented at the Electronic Imaging 2006.

Gul, G., & Kurugollu, F. (2010). SVD-Based Universal Spatial Domain Image Steganalysis. *Information Forensics and Security, IEEE Transactions on, 5*(2), 349-353. doi: 10.1109/tifs.2010.2041826

Han, Z., Fenlin, L., & Xiangyang, L. (2009, 15-18 Feb. 2009). *A wavelet-based blind JPEG image steganalysis using co-occurrence matrix.* Paper presented at the Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on.

Harmsen, J. J., & Pearlman, W. A. (2003). *Steganalysis of additive-noise modelable information hiding.* Paper presented at the Electronic Imaging 2003.

Hayati, P., Potdar, V., & Chang, E. (2007). *A survey of steganographic and steganalytic tools for the digital forensic investigator.* Paper presented at the Workshop of Information Hiding and Digital Watermarking, Moncton, New Brunswick, Canada.

Hsien-Wen, T., & Chin-Chen, C. (2004, 14-16 Sept. 2004). *Steganography using JPEG-compressed images.* Paper presented at the Computer and Information Technology, 2004. CIT '04. The Fourth International Conference on.

Huang, W., Zhao, Y., & Ni, R.-R. (2011). Block Based Adaptive Image Steganography Using LSB Matching Revisited. *Journal of Electronic Science and Technology, 9*(4), 291-296.

Iranpour, M., & Farokhian, F. (2013, 11-13 Dec. 2013). *Minimal distortion steganography using well-defined functions.* Paper presented at the High Capacity Optical Networks and Enabling Technologies (HONET-CNS), 2013 10th International Conference on.

Jain, A. K., & Uludag, U. (2002, 7-8 June). *Hiding fingerprint minutiae in images.* Paper presented at the Proceedings of 3rd Workshop on Automatic Identification Advanced Technologies, New York, USA.

Johnson, N., & Jajodia, S. (1998). Steganalysis of Images Created Using Current Steganography Software *Information Hiding* (Vol. 1525, pp. 273-289): Springer Berlin Heidelberg.

Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. *IEEE computer, 31*(2), 26-34.

Johnson, N. F., & Jajodia, S. (1998, 1-3 Sep 1998). *Steganalysis: the investigation of hidden information.* Paper presented at the Information Technology Conference, 1998. IEEE.

Judge, J. (2001). Steganography: past, present, future. SANS Institute publication. Retrieved 26/02, 2014, from http://www.sans.org/reading-room/whitepapers/stenganography/steganography-past-present-future-552?show=552.php&cat=stenganography

Katzenbeisser, S., & Petitcolas, F. A. (2000). *Information hiding techniques for steganography and digital watermarking* (Vol. 316): Artech house Norwood.

Katzenbeisser, S., & Petitolas, F. (2000). Information Hiding Techniques for Steganography and Digital Watermaking. *EDPACS, 28*(6), 1-2. doi: 10.1201/1079/43263.28.6.20001201/30373.5

Kawaguchi, E., & Eason, R. O. (1999). *Principles and applications of BPCS steganography.* Paper presented at the Photonics East (ISAM, VVDC, IEMB).

Ker, A. (2005a). A General Framework for Structural Steganalysis of LSB Replacement. In M. Barni, J. Herrera-Joancomartí, S. Katzenbeisser & F. Pérez-González (Eds.), *Information Hiding* (Vol. 3727, pp. 296-311): Springer Berlin Heidelberg.

Ker, A. (2005b). Improved Detection of LSB Steganography in Grayscale Images. In J. Fridrich (Ed.), *Information Hiding* (Vol. 3200, pp. 97-115): Springer Berlin Heidelberg.

Ker, A. D. (2004). *Quantitative evaluation of pairs and RS steganalysis.* Paper presented at the Electronic Imaging 2004.

Ker, A. D. (2005a). *Resampling and the detection of LSB matching in color bitmaps.* Paper presented at the Electronic Imaging 2005.

Ker, A. D. (2005b). Steganalysis of LSB matching in grayscale images. *Signal Processing Letters, IEEE, 12*(6), 441-444.

Ker, A. D. (2007a). *A fusion of maximum likelihood and structural steganalysis.* Paper presented at the Information Hiding.

Ker, A. D. (2007b). *Optimally weighted least-squares steganalysis.* Paper presented at the Electronic Imaging 2007.

Ker, A. D. (2007c). Steganalysis of Embedding in Two Least-Significant Bits. *Information Forensics and Security, IEEE Transactions on, 2*(1), 46-54. doi: 10.1109/tifs.2006.890519

Ker, A. D., Bas, P., Böhme, R., Cogranne, R., Craver, S., Filler, T., . . . Pevný, T. (2013). *Moving steganography and steganalysis from the laboratory into the real world*. Paper presented at the Proceedings of the first ACM workshop on Information hiding and multimedia security, Montpellier, France.

Ker, A. D., & Böhme, R. (2008). *Revisiting weighted stego-image steganalysis.* Paper presented at the Electronic Imaging 2008.

Khalind, O., & Aziz, B. (2014). *Detecting 2LSB steganography using extended pairs of values analysis.* Paper presented at the Mobile Multimedia/Image Processing, Security, and Applications 2014.

Khalind, O. S., Hernandez-Castro, J. C., & Aziz, B. (2013). A study on the false positive rate of Stegdetect. *Digital Investigation, 9*(3–4), 235-245. doi: http://dx.doi.org/10.1016/j.diin.2013.01.004

Kharrazi, M., Sencar, H., & Memon, N. (2006). Improving Steganalysis by Fusion Techniques: A Case Study with Image Steganography. In Y. Shi (Ed.), *Transactions on Data Hiding and Multimedia Security I* (Vol. 4300, pp. 123-137): Springer Berlin Heidelberg.

Kharrazi, M., Sencar, H. T., & Memon, N. (2005). *Benchmarking steganographic and steganalysis techniques.* Paper presented at the Electronic Imaging 2005.

Kipper, G. (2004). *Investigator's guide to steganography*. Florida: crc press.

Kivanc Mihcak, M., Kozintsev, I., Ramchandran, K., & Moulin, P. (1999). Low-complexity image denoising based on statistical modeling of wavelet coefficients. *Signal Processing Letters, IEEE, 6*(12), 300-303. doi: 10.1109/97.803428

Kumar, P. M., & Shunmuganathan, K. L. (2012). Developing a Secure Image Steganographic System Using TPVD Adaptive LSB Matching Revisited Algorithm for Maximizing the Embedding Rate. *Information Security Journal: A Global Perspective, 21*(2), 65-70. doi: 10.1080/19393555.2011.642063

Lee, K., Westfeld, A., & Lee, S. (2006). Category Attack for LSB Steganalysis of JPEG Images. In Y. Shi & B. Jeon (Eds.), *Digital Watermarking* (Vol. 4283, pp. 35-48): Springer Berlin Heidelberg.

Lee, Y. K., & Chen, L. H. (2000). High capacity image steganographic model. *IEE Proceedings - Vision, Image and Signal Processing, 147*(3), 288-294. http://digital-library.theiet.org/content/journals/10.1049/ip-vis_20000341

Li, B., He, J., Huang, J., & Shi, Y. Q. (2011). A survey on image steganography and steganalysis. *Journal of Information Hiding and Multimedia Signal Processing, 2*(2), 142-172.

Li, B., Huang, J., & Shi, Y. Q. (2008). *Textural features based universal steganalysis.* Paper presented at the Electronic Imaging 2008.

Li, X., Yang, B., Cheng, D., & Zeng, T. (2009). A generalization of LSB matching. *Signal Processing Letters, IEEE, 16*(2), 69-72.

Liang, G.-l., Wang, S.-z., & Zhang, X.-p. (2007). Steganography in binary image by checking data-carrying eligibility of boundary pixels. *Journal of Shanghai University (English Edition), 11*(3), 272-277. doi: 10.1007/s11741-007-0317-2

Lin, E. T., & Delp, E. J. (1999). *A review of data hiding in digital images.* Paper presented at the Image Processing, Image Quality, Image Capture Systems.

Liu, C.-L., & Liao, S.-R. (2008). High-performance JPEG steganography using complementary embedding strategy. *Pattern Recognition, 41*(9), 2945-2955. doi: http://dx.doi.org/10.1016/j.patcog.2008.03.005

Lou, D.-C., Chou, C.-L., Tso, H.-K., & Chiu, C.-C. (2012). Active steganalysis for histogram-shifting based reversible data hiding. *Optics Communications, 285*(10–11), 2510-2518. doi: http://dx.doi.org/10.1016/j.optcom.2012.01.021

Lou, D.-C., & Liu, J.-L. (2002). Steganographic method for secure communications. *Computers & Security, 21*(5), 449-460.

Lu, P., Luo, X., Tang, Q., & Shen, L. (2005). An Improved Sample Pairs Method for Detection of LSB Embedding. In J. Fridrich (Ed.), *Information Hiding* (Vol. 3200, pp. 116-127): Springer Berlin Heidelberg.

Luo, X., Liu, F., Yang, C., Lian, S., & Zeng, Y. (2012). Steganalysis of adaptive image steganography in multiple gray code bit-planes. *Multimedia Tools and Applications, 57*(3), 651-667.

Luo, X., Wang, Q., Yang, C., & Liu, F. (2006). *Detection of LTSB steganography based on quartic equation.* Paper presented at the The 8th International conference of Advanced Communication Technology.

Lyu, S., & Farid, H. (2003). Detecting Hidden Messages Using Higher-Order Statistics and Support Vector Machines. In F. P. Petitcolas (Ed.), *Information Hiding* (Vol. 2578, pp. 340-354): Springer Berlin Heidelberg.

Lyu, S., & Farid, H. (2004). *Steganalysis using color wavelet statistics and one-class support vector machines.* Paper presented at the Electronic Imaging 2004.

Maes, M. (1998). Twin Peaks: The Histogram Attack to Fixed Depth Image Watermarks *Information Hiding* (Vol. 1525, pp. 290-305): Springer Berlin Heidelberg.

Marvel, L., Boncelet, C., Jr., & Retter, C. (1998). Reliable Blind Information Hiding for Images *Information Hiding* (Vol. 1525, pp. 48-61): Springer Berlin Heidelberg.

Marvel, L. M., Boncelet, C. G., Jr., & Retter, C. T. (1999). Spread spectrum image steganography. *Image Processing, IEEE Transactions on, 8*(8), 1075-1083. doi: 10.1109/83.777088

Max, B. (2007). *Principles of Data Mining*. London: Springer London Ltd, Published.

Mielikainen, J. (2006). LSB matching revisited. *Signal Processing Letters, IEEE, 13*(5), 285-287.

Min, W., Tang, E., & Lin, B. (2000, 2000). *Data hiding in digital binary image.* Paper presented at the Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on.

Morkel, T., Eloff, J. H., & Olivier, M. S. (2005). *An overview of image steganography.* Paper presented at the ISSA.

Never-compressed image database. 2014, from http://www.shsu.edu/~qxl005/New/Downloads/index.html

Niu, C., Sun, X., Qin, J., & Xia, Z. (2009). *Steganalysis of two least significant bits embedding based on least square method.* Paper presented at the Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on.

Orebaugh, A. D. (2004). Steganalysis: A Steganography Intrusion Detection System. June 2012, from http://securityknox.com/Steg_project.pdf

Petitcolas, F. A. P., & Anderson, R. J. (1999, Jul 1999). *Evaluation of copyright marking systems.* Paper presented at the Multimedia Computing and Systems, 1999. IEEE International Conference on.

Pevny, T., & Fridrich, J. (2007). *Merging Markov and DCT features for multi-class JPEG steganalysis.* Paper presented at the Electronic Imaging 2007.

Pevný, T., & Ker, A. D. (2013). *The challenges of rich features in universal steganalysis.* Paper presented at the IS&T/SPIE Electronic Imaging.

Ponomarenko, N., Lukin, V., Egiazarian, K., Astola, J., Carli, M., & Battisti, F. (2008, 8-10 Oct. 2008). *Color image database for evaluation of image quality metrics.* Paper presented at the Multimedia Signal Processing, 2008 IEEE 10th Workshop on.

Popa, R. (1998). An analysis of steganographic techniques. *The Politehnica University of Timisoara, Faculty of Automatics and Computers, Department of Computer Science and Software Engineering*.

Provos, N. (2008). OutGuess – steganography detection. May 2012, from http://www.outguess.org/detection.php

Provos, N., & Honeyman, P. (2001). Detecting Steganographic Content on the Internet: CITI Technical Report 01-11.

Provos, N., & Honeyman, P. (2003). Hide and seek: an introduction to steganography. *Security & Privacy, IEEE, 1*(3), 32-44. doi: 10.1109/msecp.2003.1203220

Qingzhong, L., Chen, Y., & Dongsheng, C. (2006, 0-0 0). *A Robust Image Hiding Method Based on Sign Embedding and Fuzzy Classification.* Paper presented at the Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on.

Radhakrishnan, R., Kharrazi, M., & Memon, N. (2005). Data Masking: A New Approach for Steganography? *Journal of VLSI signal processing systems for signal, image and video technology, 41*(3), 293-303. doi: 10.1007/s11265-005-4153-1

Reith, M., Carr, C., & Gunsch, G. (2002). An examination of digital forensic models. *International Journal of Digital Evidence, 1*(3), 1-12.

Rinaldo, R., & Calvagno, G. (1995). Image coding by block prediction of multiresolution subimages. *Image Processing, IEEE Transactions on, 4*(7), 909-920. doi: 10.1109/83.392333

Rufeng, C., Xinggang, Y., Xiangwei, K., & Xiaohui, B. (2004, 17-21 May 2004). *A DCT-based image steganographic method resisting statistical attacks.* Paper presented at the Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on.

SALLEE, P. (2005). MODEL-BASED METHODS FOR STEGANOGRAPHY AND STEGANALYSIS. *International Journal of Image and Graphics, 05*(01), 167-189. doi: doi:10.1142/S0219467805001719

Shaou-Gang, M., Chin-Ming, H., Yuh-Show, T., & Hui-Mei, C. (2000, 2000). *A secure data hiding technique with heterogeneous data-combining capability for electronic patient records.* Paper presented at the Engineering in Medicine and Biology Society, 2000. Proceedings of the 22nd Annual International Conference of the IEEE.

Shapiro, J. M. (1993). Embedded image coding using zerotrees of wavelet coefficients. *Signal Processing, IEEE Transactions on, 41*(12), 3445-3462. doi: 10.1109/78.258085

Sharp, T. (2001). *An implementation of key-based digital signal steganography.* Paper presented at the Information Hiding.

Shi, Y., Chen, C., & Chen, W. (2007). A Markov Process Based Approach to Effective Attacking JPEG Steganography. In J. Camenisch, C. Collberg, N. Johnson & P. Sallee (Eds.), *Information Hiding* (Vol. 4437, pp. 249-264): Springer Berlin Heidelberg.

Shi, Y. Q., Guorong, X., Zou, D., Jianjiong, G., Chengyun, Y., Zhenping, Z., . . . Chunhua, C. (2005, 6-8 July 2005). *Image steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image, and neural network.* Paper presented at the Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on.

Simmons, G. (1984). The Prisoners' Problem and the Subliminal Channel. In D. Chaum (Ed.), *Advances in Cryptology* (pp. 51-67): Springer US.

Smith, J., & Comiskey, B. (1996). Modulation and information hiding in images. In R. Anderson (Ed.), *Information Hiding* (Vol. 1174, pp. 207-226): Springer Berlin Heidelberg.

Stoica, A., Vertan, C., & Fernandez-Maloigne, C. (2003, 0-0 2003). *Objective and subjective color image quality evaluation for JPEG 2000 compressed images.* Paper presented at the Signals, Circuits and Systems, 2003. SCS 2003. International Symposium on.

Sullivan, K., Madhow, U., Chandrasekaran, S., & Manjunath, B. (2006). Steganalysis for Markov cover data with applications to images. *Information Forensics and Security, IEEE Transactions on, 1*(2), 275-287.

Sun, H.-M., Chen, Y.-H., & Wang, K.-H. (2006). An image data hiding scheme being perfectly imperceptible to histogram attacks. *Image and Vision Computing New Zealand IVCNZ, 16*(1), 27-29.

Tao, Z., & Xijian, P. (2003, 6-10 April 2003). *Reliable detection of LSB steganography based on the difference image histogram.* Paper presented at the Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on.

VenkatramanS, Ajith, A., & Paprzycki, M. (2004, 5-7 April 2004). *Significance of steganography on data security.* Paper presented at the Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on.

Wang, H., & Wang, S. (2004). Cyber warfare: steganography vs. steganalysis. *Commun. ACM, 47*(10), 76-82. doi: 10.1145/1022594.1022597

Watters, P. A., Martin, F., & Stripf, S. H. (2005, 4-7 July 2005). *Visual steganalysis of LSB-encoded natural images.* Paper presented at the Information Technology and Applications, 2005. ICITA 2005. Third International Conference on.

Wayner, P. (1992). Mimic functions. *Cryptologia, 16*(3), 193-214.

Wayner, P. (2002). *Disappearing Cryptography: Information Hiding: Steganography & Watermarking* (Second Edition ed.). San Francisco, USA: Morgan Kaufmann.

Weiqi, L., Fangjun, H., & Jiwu, H. (2010). Edge Adaptive Image Steganography Based on LSB Matching Revisited. *Information Forensics and Security, IEEE Transactions on, 5*(2), 201-214. doi: 10.1109/tifs.2010.2041812

Westfeld, A. (2001). F5—A Steganographic Algorithm. In I. Moskowitz (Ed.), *Information Hiding* (Vol. 2137, pp. 289-302): Springer Berlin Heidelberg.

Westfeld, A. (2003). Detecting Low Embedding Rates. In F. P. Petitcolas (Ed.), *Information Hiding* (Vol. 2578, pp. 324-339): Springer Berlin Heidelberg.

Westfeld, A., & Pfitzmann, A. (2000). Attacks on Steganographic Systems. In A. Pfitzmann (Ed.), *Information Hiding* (Vol. 1768, pp. 61-76): Springer Berlin Heidelberg.

Westfeld, A., & Pfitzmann, A. (2001). *High capacity despite better steganalysis (F5–a steganographic algorithm).* Paper presented at the Information Hiding, 4th International Workshop.

Wu, D.-C., & Tsai, W.-H. (2003). A steganographic method for images by pixel-value differencing. *Pattern Recognition Letters, 24*(9), 1613-1626.

Wu, H.-C., Wu, N.-I., Tsai, C.-S., & Hwang, M.-S. (2005). Image steganographic scheme based on pixel-value differencing and LSB replacement methods. *IEE Proceedings-Vision, Image and Signal Processing, 152*(5), 611-615.

Wu, N.-I., & Hwang, M.-S. (2007). Data hiding: current status and key issues. *IJ Network Security, 4*(1), 1-9.

Xi, L., Ping, X., & Zhang, T. (2010). *Improved LSB matching steganography resisting histogram attacks.* Paper presented at the Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on.

Xiangyang, L., Fenlin, L., Shiguo, L., Chunfang, Y., & Gritzalis, S. (2011). On the Typical Statistic Features for Image Blind Steganalysis. *Selected Areas in Communications, IEEE Journal on, 29*(7), 1404-1422. doi: 10.1109/jsac.2011.110807

Xiaochuan, C., Yunhong, W., Tieniu, T., & Guo, L. (2006, 0-0 0). *Blind Image Steganalysis Based on Statistical Analysis of Empirical Matrix.* Paper presented at the Pattern Recognition, 2006. ICPR 2006. 18th International Conference on.

Xiaopi, Y., Yunhong, W., & Tieniu, T. (2004, 23-26 Aug. 2004). *On estimation of secret message length in JSteg-like steganography.* Paper presented at the Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on.

Xuan, G., Shi, Y., Gao, J., Zou, D., Yang, C., Zhang, Z., . . . Chen, W. (2005). Steganalysis Based on Multiple Features Formed by Statistical Moments of Wavelet Characteristic Functions. In M. Barni, J. Herrera-Joancomartí, S. Katzenbeisser & F. Pérez-González (Eds.), *Information Hiding* (Vol. 3727, pp. 262-277): Springer Berlin Heidelberg.

Yang, C., Liu, F., Luo, X., & Liu, B. (2008). Steganalysis frameworks of embedding in multiple least-significant bits. *Information Forensics and Security, IEEE Transactions on, 3*(4), 662-672.

Yu, L., Zhao, Y., Ni, R., & Li, T. (2010). Improved adaptive LSB steganography based on chaos and genetic algorithm. *EURASIP J. Adv. Signal Process, 2010*, 1-6. doi: 10.1155/2010/876946

Yu, X., & Babaguchi, N. (2008). *Weighted stego-image based steganalysis in multiple least significant bits.* Paper presented at the Multimedia and Expo, 2008 IEEE International Conference on.

Yu, X., Tan, T., & Wang, Y. (2005). *Extended optimization method of LSB steganalysis.* Paper presented at the IEEE International Conference on Image Processing, 2005. ICIP 2005. .

Yu, Y.-H., Chang, C.-C., & Hu, Y.-C. (2005). Hiding secret data in images via predictive coding. *Pattern Recognition, 38*(5), 691-705. doi: http://dx.doi.org/10.1016/j.patcog.2004.11.006

Yue, L., Chang-Tsun, L., & Chia-Hung, W. (2007, 29-31 Aug. 2007). *Protection of Mammograms Using Blind Steganography and Watermarking.* Paper presented at the Information Assurance and Security, 2007. IAS 2007. Third International Symposium on.

Zeng, W., Lin, C.-Y., & Yu, H. (2006). Multimedia Security Technologies for Digital Rights Management. San Diego, USA: Elsevier.

Zhang, J., Cox, I. J., & Doërr, G. (2007). *Steganalysis for LSB matching in images with high-frequency noise.* Paper presented at the Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on.

Zhang, K., Gao, H.-Y., & Bao, W.-s. (2009, 25-26 July 2009). *Stegananlysis Method of Two Least-Significant Bits Steganography.* Paper presented at the International Conference on Information Technology and Computer Science, 2009. ITCS 2009.

Zhang, T., & Ping, X. (2003a). *A fast and effective steganalytic technique against JSteg-like algorithms*. Paper presented at the Proceedings of the 2003 ACM symposium on Applied computing, Melbourne, Florida.

Zhang, T., & Ping, X. (2003b). A new approach to reliable detection of LSB steganography in natural images. *Signal Processing, 83*(10), 2085-2093. doi: http://dx.doi.org/10.1016/S0165-1684(03)00169-5

Zhang, W., Zhang, X., & Wang, S. (2007). A Double Layered "Plus-Minus One" Data Embedding Scheme. *Signal Processing Letters, IEEE, 14*(11), 848-851. doi: 10.1109/lsp.2007.903255

Zhou, W., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on, 13*(4), 600-612. doi: 10.1109/TIP.2003.819861

Zöllner, J., Federrath, H., Klimant, H., Pfitzmann, A., Piotraschke, R., Westfeld, A., . . . Wolf, G. (1998). *Modeling the security of steganographic systems.* Paper presented at the Information Hiding.

FORM UPR16

Research Ethics Review Checklist

University of Portsmouth

Please include this completed form as an appendix to your thesis (see the Postgraduate Research Student Handbook for more information

Postgraduate Research Student (PGRS) Information				**Student ID:**	608751
PGRS Name:	Omed Saleem Khalind				
Department:	Computing	**First Supervisor:**	Dr. Benjamin Aziz		
Start Date: (or progression date for Prof Doc students)		**1st October 2011**			
Study Mode and Route:	Part-time ☐	MPhil ☐	MD ☐		
	Full-time ☒	PhD ☒	Professional Doctorate ☐		

Title of Thesis:	New Methods to Improve the Pixel Domain Steganography, Steganalysis, and Simplify the Assessment of Steganalysis Tools
Thesis Word Count: (excluding ancillary data)	49532

If you are unsure about any of the following, please contact the local representative on your Faculty Ethics Committee for advice. Please note that it is your responsibility to follow the University's Ethics Policy and any relevant University, academic or professional guidelines in the conduct of your study

Although the Ethics Committee may have given your study a favourable opinion, the final responsibility for the ethical conduct of this work lies with the researcher(s).

UKRIO Finished Research Checklist:
(If you would like to know more about the checklist, please see your Faculty or Departmental Ethics Committee rep or see the online version of the full checklist at: http://www.ukrio.org/what-we-do/code-of-practice-for-research/)

a) Have all of your research and findings been reported accurately, honestly and within a reasonable time frame?	YES ☒ NO ☐
b) Have all contributions to knowledge been acknowledged?	YES ☒ NO ☐
c) Have you complied with all agreements relating to intellectual property, publication and authorship?	YES ☒ NO ☐
d) Has your research data been retained in a secure and accessible form and will it remain so for the required duration?	YES ☒ NO ☐
e) Does your research comply with all legal, ethical, and contractual requirements?	YES ☒ NO ☐

Candidate Statement:

I have considered the ethical dimensions of the above named research project, and have successfully obtained the necessary ethical approval(s)

Ethical review number(s) from Faculty Ethics Committee (or from NRES/SCREC):	C302-80FD-1E90-E362-8F78-4377-BE12-4D54

If you have *not* submitted your work for ethical review, and/or you have answered 'No' to one or more of questions a) to e), please explain below why this is so:

Signed *(PGRS)*:		**Date:** 6th September 2015